ADOBE® FLASH® CS3 PROFESSIONAL

USER GUIDE

Using Adobe® Flash® CS3 Professional for Windows® and Macintosh

Adobe Systems Incorporated, 345 Park Avenue, San Jose, California 95110, USA.

Contents

Chapter 1: Getting started

Chapter 2: Workspace

Chapter 3: Creating and managing documents

Chapter 4: Adobe Version Cue

Chapter 5: Using imported artwork

Chapter 6: Drawing

Chapter 7: Working with color, strokes, and fills

Chapter 8: Working with graphic objects

Chapter 9: Using symbols, instances, and library assets

Chapter 10: Creating animation

Chapter 11: Special effects

Chapter 12: Working with text

Chapter 13: Creating multilanguage text

Chapter 14: Working with sound

Chapter 15: Working with video

Chapter 16: Creating e-learning content

Chapter 17: Creating accessible content

Chapter 18: Working with screens

Chapter 19: ActionScript

Chapter 20: Publishing Flash content

Chapter 21: Exporting from Flash

Chapter 22: Printing with Flash

Chapter 23: Best practices

Chapter 1: Getting started

If you haven't installed your new software, begin by reading some information on installation and other preliminaries. Before you begin working with your software, take a few moments to read an overview of Adobe® Help and of the many resources available to users. You have access to instructional videos, plug-ins, templates, user communities, seminars, tutorials, RSS feeds, and much more.

Installation

Requirements

❖ To review complete system requirements and recommendations for your Adobe® software, see the Read Me file on the installation DVD.

Install the software

1 Close any other Adobe applications open on your computer.

2 Insert the installation disc into your DVD drive, and follow the on-screen instructions.

Note: *For more information, see the Read Me file on the installation DVD.*

Activate the software

If you have a single-user retail license for your Adobe software, you will be asked to activate your software; this is a simple, anonymous process that you must complete within 30 days of starting the software.

For more information on product activation, see the Read Me file on your installation DVD, or visit the Adobe website at www.adobe.com/go/activation.

1 If the Activation dialog box isn't already open, choose Help > Activate.

2 Follow the on-screen instructions.

Note: *If you want to install the software on a different computer, you must first deactivate it on your computer. Choose Help > Deactivate.*

Register

Register your product to receive complimentary installation support, notifications of updates, and other services.

❖ To register, follow the on-screen instructions in the Registration dialog box, which appears after you install and activate the software.

💡 *If you postpone registration, you can register at any time by choosing Help > Registration.*

Change or reinstall Flash Player

1 Close your browser.

2 Remove any currently installed version of the player.

For instructions, see TechNote 14157 on the Adobe® Flash® Support Center at www.adobe.com/go/tn_14157.

3 To begin installation, run one of the following in your Players folder:

- For the ActiveX control for Windows® (Internet Explorer or AOL), run the Install Flash Player 9 AX.exe file.
- For the plug-in for Windows (CompuServe, Firefox, Mozilla, Netscape, or Opera), run the Install Flash Player 9.exe file.
- For the plug-in for Macintosh® (AOL, CompuServe, Firefox, Internet Explorer for Macintosh, Netscape, Opera, or Safari), run Install Flash Player 9 (Mac OS 9.x) or Install Flash Player 9 OS X (Mac OS X.x).

Note: *To verify installation in Netscape, select Help > About Plug-ins from within the browser.*

Using Help

About Flash Help

The Flash Help panel (Help > Flash Help) contains the full set of user-assistance information provided with Flash. To view a Help topic, click its title in the table of contents. Above the topic, you can see its relative location in the hierarchy of topics.

You can hide the table of contents. To display it again, click the Table of Contents button 📖. When you search Help, the returned topics take the place of the table of contents. To redisplay the table of contents, click Clear.

The Help panel also displays context-sensitive reference information that you access from the Actions panel.

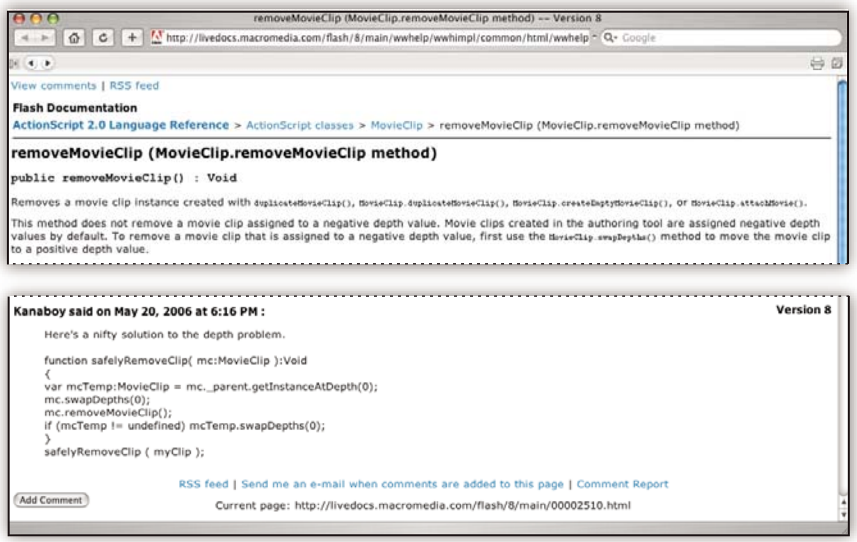
Adobe Help resources

Documentation for your Adobe software is available in a variety of formats.

In-product and LiveDocs Help

In-product Help provides access to all documentation and instructional content available at the time the software ships. It is available through the Help menu in your Adobe software.

LiveDocs Help includes all the content from in-product Help, plus updates and links to additional instructional content available on the web. For some products, you can also add comments to the topics in LiveDocs Help. Find LiveDocs Help for your product in the Adobe Help Resource Center, at www.adobe.com/go/documentation.


removeMovieClip (MovieClip.removeMovieClip method) -- Version 8
http://livedocs.macromedia.com/flash/8/main/wwhelp/wwhimpl/common/html/wwhelp
Google
View comments | RSS feed
Add Comment

Flash Documentation

ActionScript 2.0 Language Reference > ActionScript classes > MovieClip > removeMovieClip (MovieClip.removeMovieClip method)

removeMovieClip (MovieClip.removeMovieClip method)

`public removeMovieClip() : Void`

Removes a movie clip instance created with `duplicateMovieClip()`, `MovieClip.duplicateMovieClip()`, `MovieClip.createEmptyMovieClip()`, or `MovieClip.attachMovie()`.

This method does not remove a movie clip assigned to a negative depth value. Movie clips created in the authoring tool are assigned negative depth values by default. To remove a movie clip that is assigned to a negative depth value, first use the `MovieClip.swapDepths()` method to move the movie clip to a positive depth value.

Kanaboy said on May 20, 2006 at 6:16 PM : Version 8

Here's a nifty solution to the depth problem.

```
function safelyRemoveClip( mc:MovieClip ):Void
{
var mcTemp:MovieClip = mc._parent.getInstanceAtDepth(0);
mc.swapDepths(0);
mc.removeMovieClip();
if (mcTemp != undefined) mcTemp.swapDepths(0);
}
safelyRemoveClip ( myClip );
```

RSS feed | Send me an e-mail when comments are added to this page | Comment Report

Current page: http://livedocs.macromedia.com/flash/8/main/00002510.html

Most versions of in-product and LiveDocs Help let you search across the Help systems of multiple products. Topics may also contain links to relevant content on the web or to topics in the Help of another product.

Think of Help, both in the product and on the web, as a hub for accessing additional content and communities of users. The most complete and up-to-date version of Help is always on the web.

PDF documentation

The in-product Help is also available as a PDF that is optimized for printing. Other documents, such as installation guides and white papers, may also be provided as PDFs.

All PDF documentation is available through the Adobe Help Resource Center, at www.adobe.com/go/documentation. To see the PDF documentation included with your software, look in the Documents folder on the installation or content DVD.

Printed documentation

Printed editions of the in-product Help are available for purchase in the Adobe Store, at www.adobe.com/go/store. You can also find books published by Adobe publishing partners in the Adobe Store.

A printed workflow guide is included with all Adobe Creative Suite® 3 products, and stand-alone Adobe products may include a printed getting started guide.

Searching Flash Help

Flash can search all Flash Help systems or a single Help system (such as *Using Flash*).

You can also search the text of a single topic: Click in the topic to give it focus and press Ctrl+F (Windows) or Command+F (Macintosh).

You can search Flash Help for a combination of words and phrases:

Single-word searches Return a list of help pages that contain the specified word. For example, if you type `timeline` in the search text field, Flash returns a list of help pages that contain the word *timeline* or *Timeline*.

Multiple-word searches Return a list of help pages that each contain all of the search terms you enter. In this case, the word *and* is implicit in the search. For example, if you type `movie clip` in the search text field, Flash returns a list of pages that contain both *movie* and *clip*—that is, *clip movie, movie clip,movie...clip,* and so on.

Explicit AND/OR searches Use the words *and* or *or* to refine the search results. For example, if you type `timeline and keyframe or tween` in the search text field, Flash returns a list of help pages that contain *timeline* and *keyframe* and help pages that contain *timeline* and *tween*.

Exact phrase searches Use quotation marks to return only pages that contain the specific phrase you enter. For example, if you type `"motion tween"` in the search text field, Flash returns a list of help pages that contain the phrase *motion tween*, but not pages that contain separate instances of *motion* and *tween*.

Exact phrase with explicit AND/OR searches Use a combination of quotation marks and the words *and* or *or* to further refine your searches. For example, if you type `"motion tween" and "ActionScript"` in the search field, Flash returns a list of pages that contain both the phrase *motion tween* and the word *ActionScript*.

Access context-sensitive Help from the Actions panel

1 To select an item for reference, do any of the following:

- Select an item in the Actions panel toolbox pane (on the left side of the Actions panel).
- Select an ActionScript term in the Actions panel in the Script pane.
- Place the insertion point before an ActionScript term in the Actions panel in the Script pane.

2 To open the Help panel reference page for the selected item, do one of the following:

- Press F1.
- Right-click the item and select View Help.
- Click Help ? above the Script pane.

Choosing the right Help documents

Flash Help contains many documents. The following list describes each document's purpose and contents:

- *Using Flash* contains an introduction to what Flash is, what you can do with it, and how the Flash user interface works. It also contains detailed information about using all of the tools and features in the Flash authoring tool.
- *Programming ActionScript 3.0* provides a detailed description of the ActionScript 3.0 language, intended for beginning and experienced scripters. *Programming ActionScript 3.0* explains the basic concepts of writing code, including how to use logic to write code that makes decisions, how to make your Flash projects respond to user actions, and how to write code to perform the most common tasks in Flash. ActionScript 3.0 is faster and more appropriate for computationally intensive applications than ActionScript 2.0, and is somewhat more complex than ActionScript 2.0.
- The *ActionScript 3.0 Language and Components Reference* includes dictionary-style entries for all of the actions, methods, and properties in the ActionScript 3.0 application programming interface (API), as well as the APIs for the ActionScript 3.0 components included with Flash. This reference is a fast way to find specific ActionScript terms to accomplish specific tasks. Each entry includes details of the term's syntax and functionality, and code examples.

- *Using ActionScript 3.0 Components* contains information on using and configuring ActionScript 3.0 components in a Flash document. *Components* are reusable user interface elements such as buttons, menus, and so on, that you can use in your own projects without having to create and script them yourself. Some components do not have a visual presence, but instead help you store and manage data for your application. This document also contains information about creating your own reusable components with ActionScript 3.0.
- *Learning ActionScript 2.0 in Adobe Flash* provides a detailed description of the ActionScript 2.0 language, intended for both new and more experienced scripters. *Learning ActionScript 2.0 in Adobe Flash* describes the basic concepts of writing code, including which scripts you can use in Flash, when to use each type, how to use logic to write code that makes decisions, how to make your Flash projects respond to user actions, and how to write specific code to perform the most common tasks in Flash.
- The *ActionScript 2.0 Language Reference* includes dictionary-style entries for all of the actions, methods, and properties in the ActionScript 2.0 application programming interface (API). This reference is a fast way to find specific ActionScript terms to accomplish specific tasks. Each entry includes details of the term's syntax and functionality, as well as code examples.
- *Using ActionScript 2.0 Components* contains information on using and configuring components in a Flash document. *Components* are reusable user interface elements such as buttons, menus, and so on, that you can use in your own projects without having to create and script them yourself. Some components do not have a visual presence, but instead help you store and manage data for your application. These documents also contain information about creating your own reusable components with ActionScript.
- *ActionScript 2.0 Components Language Reference* includes dictionary-style entries for all of the methods and properties that are available to each component included with Flash. You control the behavior of components with these APIs. After you understand the basics of how to use components, this reference is a fast way to find specific APIs that can help you accomplish specific tasks.
- *Extending Flash* describes how to add functionality and automation to the Flash authoring tool with custom JavaScript APIs created for that purpose.
- *Getting Started with Flash Lite 2.x* provides an introduction to the process of developing content with Adobe® Flash® Lite™ 2.x for delivery on mobile phones and devices. Flash Lite 2.x supports a subset of ActionScript 2.0.
- *Developing Flash Lite 2.x Applications* provides techniques and guidelines for creating content and applications for Flash Lite 2.x, the most current version of Adobe® Flash® Player designed for mobile phones and other devices. Because Flash Lite 2.x supports different features than the desktop version of Flash Player, techniques for creating content for Flash Lite are different from techniques for creating Flash desktop content.
- *Introduction to Flash Lite 2.x ActionScript* describes in detail the ActionScript features available in Flash Lite 2.x and explains how to accomplish common scripting tasks when using Flash Lite 2.x.
- *Flash Lite 2.x ActionScript Language Reference* provides dictionary-style entries for all of the actions, methods, and properties available in Flash Lite 2.x. Each entry includes the details of the term's syntax and functionality, as well as sample code.
- *Getting Started with Flash Lite 1.x* provides an introduction to the process of developing content with Flash Lite 1.x for delivery on mobile phones and devices. Flash Lite 1.x supports a subset of ActionScript 1.0.
- *Developing Flash Lite 1.x Applications* provides techniques and guidelines for creating content and applications for Flash Lite 1.x, the first version of Flash Player designed for mobile phones and other devices. Because Flash Lite 1.x supports different features than the desktop version of Flash Player, techniques for creating content for Flash Lite 1.x are different from techniques for creating Flash desktop content.
- *Learning Flash Lite 1.x ActionScript* describes in detail the ActionScript features available in Flash Lite 1.0 and 1.1 and explains how to perform common scripting tasks when using Flash Lite 1.x.

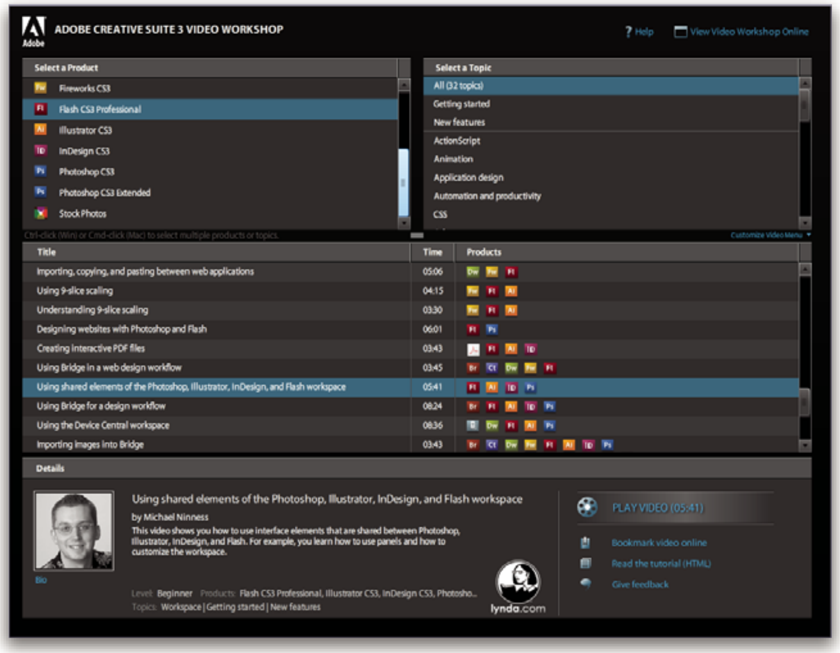
- *Flash Lite 1.x ActionScript Language Reference* provides dictionary-style entries for all of the actions, methods, and properties available in Flash Lite 1.0 and 1.1. Each entry includes the details of the term's syntax and functionality, as well as sample code.

Resources

Adobe Video Workshop

The Adobe Creative Suite 3 Video Workshop offers over 200 training videos covering a wide range of subjects for print, web, and video professionals.

You can use the Adobe Video Workshop to learn about any Creative Suite 3 product. Many videos show you how to use Adobe applications together.



Title	Time	Products
Importing, copying, and pasting between web applications	05:06	Dw Fw Fl
Using 9-slice scaling	04:15	Fw Fl Ai
Understanding 9-slice scaling	03:30	Fw Fl Ai
Designing websites with Photoshop and Flash	06:01	Fl Ps
Creating interactive PDF files	03:43	Fl Ai ID
Using Bridge in a web design workflow	03:45	Br Ct Dw Fw Fl
Using shared elements of the Photoshop, Illustrator, InDesign, and Flash workspace	05:41	Fl Ai ID Ps
Using Bridge for a design workflow	08:24	Br Fl Ai ID Ps
Using the Device Central workspace	08:36	Dw Fl Ai Ps
Importing images into Bridge	03:43	Br Ct Dw Fw Fl Ai ID Ps

Using shared elements of the Photoshop, Illustrator, InDesign, and Flash workspace

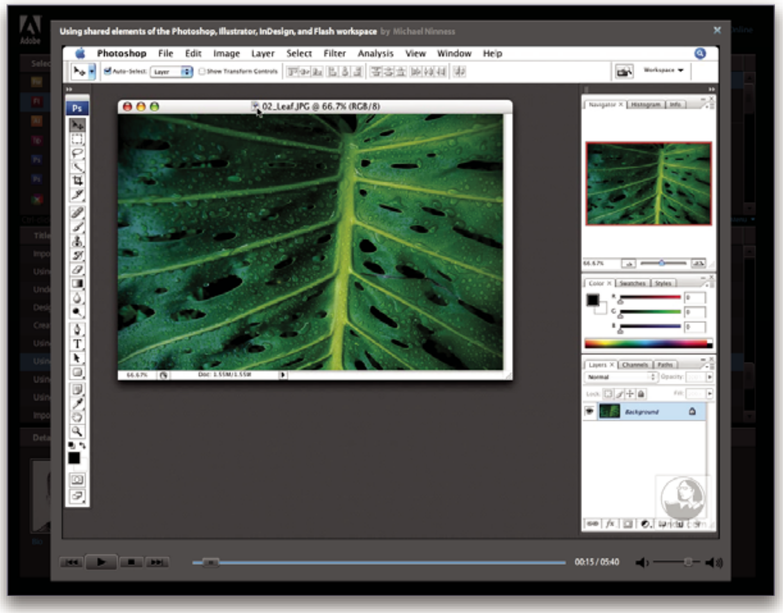
by Michael Ninness

This video shows you how to use interface elements that are shared between Photoshop, Illustrator, InDesign, and Flash. For example, you learn how to use panels and how to customize the workspace.

Level: Beginner Products: Flash CS3 Professional, Illustrator CS3, InDesign CS3, Photosho...

Topics: Workspace | Getting started | New features

When you start the Adobe Video Workshop, you choose the products you want to learn and the subjects you want to view. You can see details about each video to focus and direct your learning.


Using shared elements of the Photoshop, Illustrator, InDesign, and Flash workspace by Michael Ninness
Photoshop File Edit Image Layer Select Filter Analysis View Window Help
02_Leaf.JPG @ 66.7% (RGB/8)
Navigator Histogram Info
Color Swatches Styles
Layers Channels Paths
Background
00:15 / 05:40

Community of presenters

With this release, Adobe Systems invited the community of its users to share their expertise and insights. Adobe and lynda.com present tutorials, tips, and tricks from leading designers and developers such as Joseph Lowery, Katrin Eismann, and Chris Georgenes. You can see and hear Adobe experts such as Lynn Grillo, Greg Rewis, and Russell Brown. In all, over 30 product experts share their knowledge.

Tutorials and source files

The Adobe Video Workshop includes training for novices and experienced users. You'll also find videos on new features and key techniques. Each video covers a single subject and typically runs about 3-5 minutes. Most videos come with an illustrated tutorial and source files, so you can print detailed steps and try the tutorial on your own.

Using Adobe Video Workshop

You can access Adobe Video Workshop using the DVD included with your Creative Suite 3 product. It's also available online at www.adobe.com/go/learn_videotutorials. Adobe will regularly add new videos to the online Video Workshop, so check in to see what's new.

Flash CS3 Professional videos

Adobe Video Workshop covers a wide range of subjects for Adobe Flash® CS3 Professional, including these:

- Drawing with the Pen tool
- Creating animations using motion tweens
- Creating and animating masks
- Getting started with ActionScript 3.0

- Using the Flash Video Encoder

Videos also show you how to use Flash CS3 with other Adobe components:

- Using symbols effectively between Illustrator® and Flash
- Understanding the Fireworks® and Flash workflow
- Designing websites with Flash and Photoshop
- Creating mobile content in Flash

To access Adobe Creative Suite 3 video tutorials, visit Adobe Video Workshop at www.adobe.com/go/learn_videotutorials.

Extras

You have access to a wide variety of resources that will help you make the most of your Adobe software. Some of these resources are installed on your computer during the setup process; additional helpful samples and documents are included on the installation or content DVD. Unique extras are also offered online by the Adobe Exchange community, at www.adobe.com/go/exchange.

Installed resources

During software installation, a number of resources are placed in your application folder. To view those files, navigate to the application folder on your computer.

- Windows®: *[startup drive]*/Program files/Adobe/Adobe *[application]*
- Mac OS®: *[startup drive]*/Applications/Adobe *[application]*

The application folder may contain the following resources:

Plug-ins Plug-in modules are small software programs that extend or add features to your software. Once installed, plug-in modules appear as options in the Import or Export menu; as file formats in the Open, Save As, and Export Original dialog boxes; or as filters in the Filter submenus. For example, a number of special effects plug-ins are automatically installed in the Plug-ins folder inside the Photoshop CS3 folder.

Presets Presets include a wide variety of useful tools, preferences, effects, and images. Product presets include brushes, swatches, color groups, symbols, custom shapes, graphic and layer styles, patterns, textures, actions, workspaces, and more. Preset content can be found throughout the user interface. Some presets (for example, Photoshop Brush libraries) become available only when you select the corresponding tool. If you don't want to create an effect or image from scratch, go to the preset libraries for inspiration.

Templates Template files can be opened and viewed from Adobe Bridge, opened from the Welcome Screen, or opened directly from the File menu. Depending on the product, template files range from letterheads, newsletters, and websites to DVD menus and video buttons. Each template file is professionally constructed and represents a best-use example of product features. Templates can be a valuable resource when you need to jump-start a project.

Samples Sample files include more complicated designs and are a great way to see new features in action. These files demonstrate the range of creative possibilities available to you.

Fonts Several OpenType® fonts and font families are included with your Creative Suite product. Fonts are copied to your computer during installation:

- Windows: *[startup drive]*/Program Files/Common Files/Adobe/Fonts
- Mac OS X: *[startup drive]*/Library/Application Support/Adobe/Fonts

For information about installing fonts, see the Read Me file on the installation DVD.

DVD content

The installation or content DVD included with your product contains additional resources for use with your software. The Goodies folder contains product-specific files such as templates, images, presets, actions, plug-ins, and effects, along with subfolders for Fonts and Stock Photography. The Documentation folder contains a PDF version of the Help, technical information, and other documents such as specimen sheets, reference guides, and specialized feature information.

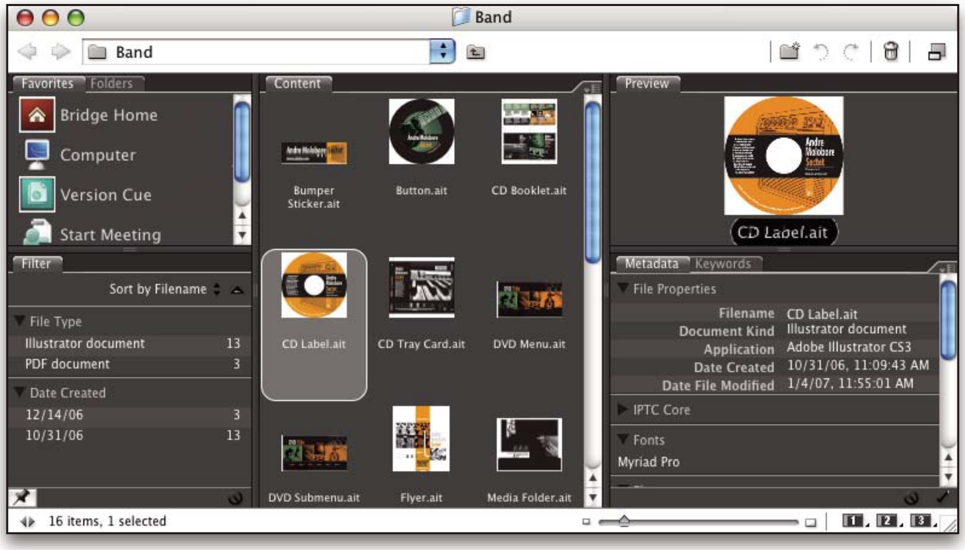
Adobe Exchange

For more free content, visit www.adobe.com/go/exchange, an online community where users download and share thousands of free actions, extensions, plug-ins, and other content for use with Adobe products.

Bridge Home

Bridge Home, a new destination in Adobe Bridge CS3, provides up-to-date information on all your Adobe Creative Suite 3 software in one convenient location. Start Adobe Bridge, then click the Bridge Home icon at the top of the Favorites panel to access the latest tips, news, and resources for your Creative Suite tools.

Note: Bridge Home may not be available in all languages.



File Type

Illustrator document	13
PDF document	3

Date Created

12/14/06	3
10/31/06	13

File Properties

Filename	CD Label.ait
Document Kind	Illustrator document
Application	Adobe Illustrator CS3
Date Created	10/31/06, 11:09:43 AM
Date File Modified	1/4/07, 11:55:01 AM

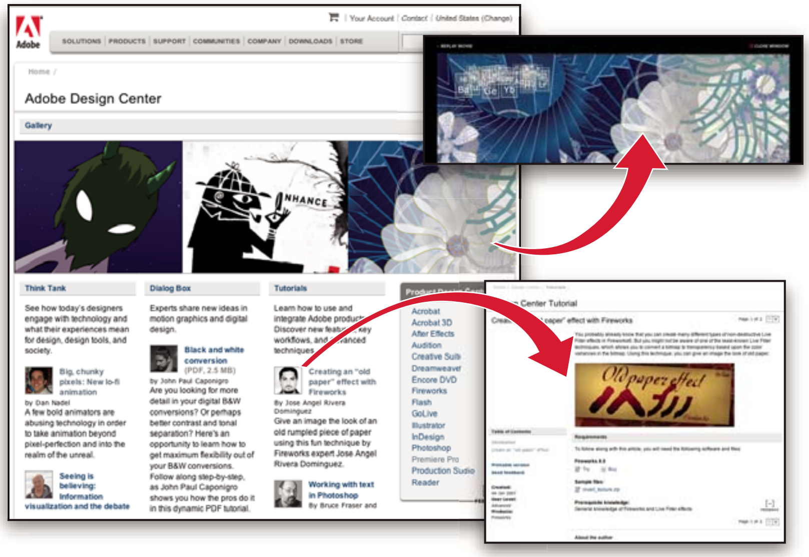
IPTC Core

Fonts

Myriad Pro

Adobe Design Center

Adobe Design Center offers articles, inspiration, and instruction from industry experts, top designers and Adobe publishing partners. New content is added monthly.



You can find hundreds of tutorials for design products and learn tips and techniques through videos, HTML tutorials, and sample book chapters.

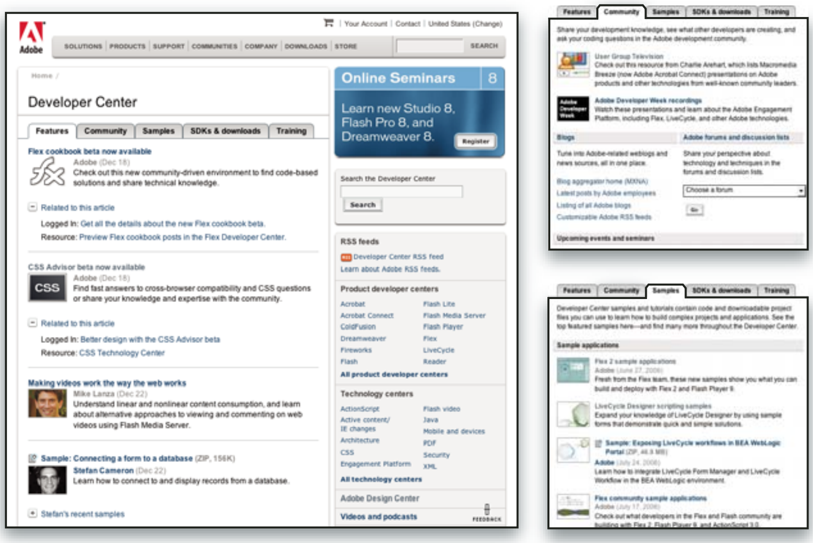
New ideas are the heart of Think Tank, Dialog Box, and Gallery:

- Think Tank articles consider how today's designers engage with technology and what their experiences mean for design, design tools, and society.
- In Dialog Box, experts share new ideas in motion graphics and digital design.
- The Gallery showcases how artists communicate design in motion.

Visit Adobe Design Center at www.adobe.com/designcenter.

Adobe Developer Center

Adobe Developer Center provides samples, tutorials, articles, and community resources for developers who build rich Internet applications, websites, mobile content, and other projects using Adobe products. The Developer Center also contains resources for developers who develop plug-ins for Adobe products.


Adobe
Your Account | Contact | United States (Change)
SOLUTIONS | PRODUCTS | SUPPORT | COMMUNITIES | COMPANY | DOWNLOADS | STORE
SEARCH
Home /
Features
Community
Samples
SDKs & downloads
Training
Search
Register
Go

Developer Center

Flex cookbook beta now available
Adobe (Dec 18)
Check out this new community-driven environment to find code-based solutions and share technical knowledge.

Related to this article
Logged In: Get all the details about the new Flex cookbook beta.
Resource: Preview Flex cookbook posts in the Flex Developer Center.

CSS Advisor beta now available
Adobe (Dec 18)
Find fast answers to cross-browser compatibility and CSS questions or share your knowledge and expertise with the community.

Related to this article
Logged In: Better design with the CSS Advisor beta
Resource: CSS Technology Center

Making videos work the way the web works
Mike Lanza (Dec 22)
Understand linear and nonlinear content consumption, and learn about alternative approaches to viewing and commenting on web videos using Flash Media Server.

Sample: Connecting a form to a database (ZIP, 156K)
Stefan Cameron (Dec 22)
Learn how to connect to and display records from a database.

Stefan's recent samples

Online Seminars 8
Learn new Studio 8, Flash Pro 8, and Dreamweaver 8.

Search the Developer Center

RSS feeds
Developer Center RSS feed
Learn about Adobe RSS feeds.

Product developer centers

Acrobat	Flash Lite
Acrobat Connect	Flash Media Server
ColdFusion	Flash Player
Dreamweaver	Flex
Fireworks	LiveCycle
Flash	Reader

All product developer centers

Technology centers

ActionScript	Flash video
Active content/ IE changes	Java
	Mobile and devices
Architecture	PDF
CSS	Security
Engagement Platform	XML

All technology centers

Adobe Design Center

Videos and podcasts

FEEDBACK

Share your development knowledge, see what other developers are creating, and ask your coding questions in the Adobe development community.

User Group Television
Check out this resource from Charlie Arehart, which lists Macromedia Breeze (now Adobe Acrobat Connect) presentations on Adobe products and other technologies from well-known community leaders.

Adobe Developer Week
Adobe Developer Week recordings
Watch these presentations and learn about the Adobe Engagement Platform, including Flex, LiveCycle, and other Adobe technologies.

Blogs
Tune into Adobe-related weblogs and news sources, all in one place.
Blog aggregator home (MXNA)
Latest posts by Adobe employees
Listing of all Adobe blogs
Customizable Adobe RSS feeds

Adobe forums and discussion lists
Share your perspective about technology and techniques in the forums and discussion lists.
Choose a forum

Upcoming events and seminars

Developer Center samples and tutorials contain code and downloadable project files you can use to learn how to build complex projects and applications. See the top featured samples here—and find many more throughout the Developer Center.

Sample applications

Flex 2 sample applications
Adobe (June 27, 2006)
Fresh from the Flex team, these new samples show you what you can build and deploy with Flex 2 and Flash Player 9.

LiveCycle Designer scripting samples
Expand your knowledge of LiveCycle Designer by using sample forms that demonstrate quick and simple solutions.

Sample: Exposing LiveCycle workflows in BEA WebLogic Portal (ZIP, 46.9 MB)
Adobe (July 24, 2006)
Learn how to integrate LiveCycle Form Manager and LiveCycle Workflow in the BEA WebLogic environment.

Flex community sample applications
Adobe (July 17, 2006)
Check out what developers in the Flex and Flash community are building with Flex 2, Flash Player 9, and ActionScript 3.0.

In addition to sample code and tutorials, you'll find RSS feeds, online seminars, SDKs, scripting guides, and other technical resources.

Visit Adobe Developer Center at www.adobe.com/go/developer.

Customer support

Visit the Adobe Support website, at www.adobe.com/support, to find troubleshooting information for your product and to learn about free and paid technical support options. Follow the Training link for access to Adobe Press books, a variety of training resources, Adobe software certification programs, and more.

Downloads

Visit www.adobe.com/go/downloads to find free updates, tryouts, and other useful software. In addition, the Adobe Store (at www.adobe.com/go/store) provides access to thousands of plug-ins from third-party developers, helping you to automate tasks, customize workflows, create specialized professional effects, and more.

Adobe Labs

Adobe Labs gives you the opportunity to experience and evaluate new and emerging technologies and products from Adobe.

At Adobe Labs, you have access to resources such as these:

- Prerelease software and technologies
- Code samples and best practices to accelerate your learning
- Early versions of product and technical documentation
- Forums, wiki-based content, and other collaborative resources to help you interact with like-minded developers

Adobe Labs fosters a collaborative software development process. In this environment, customers quickly become productive with new products and technologies. Adobe Labs is also a forum for early feedback, which the Adobe development teams use to create software that meets the needs and expectations of the community.

Visit Adobe Labs at www.adobe.com/go/labs.

User communities

User communities feature forums, blogs, and other avenues for users to share technologies, tools, and information. Users can ask questions and find out how others are getting the most out of their software. User-to-user forums are available in English, French, German, and Japanese; blogs are posted in a wide range of languages.

To participate in forums or blogs, visit www.adobe.com/communities.

What’s new

New features

The following features are new to Adobe® Flash® CS3 Professional.

CS3 Interface

The Flash user interface is updated to share a common interface with other Adobe Creative Suite CS3 components. A consistent appearance across all Adobe software helps users work more easily with multiple applications. See “Workspace” on page 15.

Adobe Bridge and Version Cue

Organize and browse Flash and other creative assets using Adobe Bridge, an independent file-management system that you can launch from within Flash. Through Adobe Bridge, you can automate workflows across Adobe Creative Suite components, apply consistent color settings across Adobe software, and access version control features and online stock photo purchase services. A Welcome screen provides centralized control of settings, as well as ongoing access to tips and tutorials in Adobe Design Center. See “Adobe Version Cue” on page 82.

Bitmap Symbol Library Item dialog box

The Bitmap Symbol Library Item dialog box has been enlarged to provide a larger preview of the bitmap. See “Using symbols, instances, and library assets” on page 207.

Multicolored bounding boxes

You can change the selection color of specific types of elements to identify each element easily. See “Get information about instances on the Stage” on page 215.

Adobe Device Central

A new way to test content created with Adobe products on emulated mobile devices, Device Central lets you select a target device from the beginning of the development process, and gives you a clear idea of what a device’s limitations are. See “Developing applications for mobile devices” on page 431.

Active content detections

To eliminate the need to first activate Flash Player so that users can interact with Flash content, Flash publishes HTML templates that you can use to embed Flash SWF files. Using these templates, embedded SWF files are activated seamlessly without the need for an additional mouse click or other user activation. See “Publishing Flash documents” on page 418.

9-slice onstage preview

Because 9-slice scaling now provides onstage preview, you can see changes and adjustments to 9-slice scaled movie clips on stage. See “About 9-slice scaling and movie clip symbols” on page 222.

Filter copy and paste

You can now copy and paste graphic filter settings from one instance to another. See “Apply filters” on page 250.

Copy and paste motion

Copy and paste motion lets you copy a motion tween and paste (or apply) the frames, tween, and symbol information to another object. When pasting the motion tween to another object, you can choose to paste all properties associated with the motion tween, or choose specific properties to apply to the other object. See “Copy and paste a motion tween” on page 232.

Copy motion as ActionScript 3.0

In addition to copying the properties of one motion tween and applying those properties to another object, you can copy the properties that define a motion tween in the Timeline as ActionScript 3.0 and apply that motion to another symbol, either in the Actions panel or in the source files (such as class files) for a Flash document that uses ActionScript 3.0. See “Copy motion as ActionScript” on page 233.

Pen tool enhancements

The Pen tool has been improved.

- The Pen tool now behaves similarly to the Illustrator Pen tool to provide a more consistent user experience across Adobe software
- The cubic-to-quadratic conversion is now more efficient, resulting in better accuracy and fewer points.

See “Drawing with the Pen tool” on page 172.

Adobe Photoshop import

You can now import Adobe Photoshop PSD files directly into Flash documents. Most Photoshop data types are supported, and several import options are provided so that you can find the best balance of image fidelity and editability within Flash. See “Import Photoshop PSD files” on page 149.

Adobe Illustrator import

You can now import Adobe Illustrator AI files directly into Flash documents. Most Illustrator data types are supported, and several import options are provided so that you can find the best balance of image fidelity and editability within Flash. See "Import Adobe Illustrator files" on page 140.

Primitive Rectangle and Oval drawing tools

New Rectangle and Oval drawing tools let you create rectangles and ovals whose properties (such as stroke or corner radius) you can edit at any time in the Property inspector. See "Draw rectangles and ovals" on page 166.

Enhanced Quicktime video support

QuickTime export is intended for users who want to distribute Flash content, such as animation, in the QuickTime video format. This release improves the quality of the exported QuickTime video file, which you can distribute as streaming video or on a DVD, or import into a video-editing application such as Adobe® Premiere®. See "Exporting QuickTime" on page 453.

Save and load cue points for Flash video

Save and load functionality has been added to the Cue Points tab to allow you to save the cue points added to one file and apply them to another. You can generate a cue points XML file based on known time codes and import it into the encoder before encoding, eliminating the need to manually add each cue point through the Flash Video Encoder user interface. See Flash Video Encoder Help.

Script Assist mode for ActionScript 3.0

Script Assist mode has been updated to include support for ActionScript 3.0. See "Script Assist mode and behaviors" on page 386.

Improvements in ActionScript

Flash has a new, improved version of ActionScript. ActionScript 3.0 offers a robust programming model familiar to developers with a basic knowledge of object-oriented programming. ActionScript 3.0 facilitates the creation of highly complex applications with large data sets and object-oriented, reusable code bases. While ActionScript 3.0 is not required for content that runs in Adobe Flash Player 9, it allows performance improvements that are available only with the new ActionScript Virtual Machine (AVM2). ActionScript 3.0 code can execute up to ten times faster than legacy ActionScript code.

The older version of ActionScript Virtual Machine, AVM1, executes ActionScript 1.0 and ActionScript 2.0 code. Flash Player 9 supports AVM1 for backward compatibility with existing and legacy content.

To learn about ActionScript 3.0, see *Programming ActionScript 3.0*.

Chapter 2: Workspace

The Adobe® Flash® CS3 Professional workspace includes tools and panels that help you create and navigate your documents. Understanding these tools will help you maximize the application's capabilities.

Flash workflow and workspace

General Flash workflow

To build a Flash application, you typically perform the following basic steps:

Plan the application.
Decide which basic tasks the application will perform.

Add media elements.
Create and import media elements, such as images, video, sound, text.

Arrange the elements.
Arrange the media elements on the Stage and in the Timeline to define when and how they appear in your application.

Apply special effects.
Apply graphic filters (such as blurs, glows, and bevels), blends, and other special effects as you see fit.

Use ActionScript to control behavior.
Write ActionScript code to control how the media elements behave, including how the elements respond to user interactions.

Test and publish your application.
Test to verify that your application is working as you intended, and find and fix any bugs you encounter. You should test the application throughout the creation process. Publish your FLA file as a SWF file that can be displayed in a web page and played back with Flash Player.

Depending on your project and your working style, you might use these steps in a different order.

For video tutorials about the Flash workflow, see the following:

- Flash workflow: www.adobe.com/go/vid0132
- Creating your first interactive Flash file: www.adobe.com/go/vid0118

For a text tutorial about creating an application, see Create an Application on the Flash Tutorials page at www.adobe.com/go/learn_fl_tutorials.

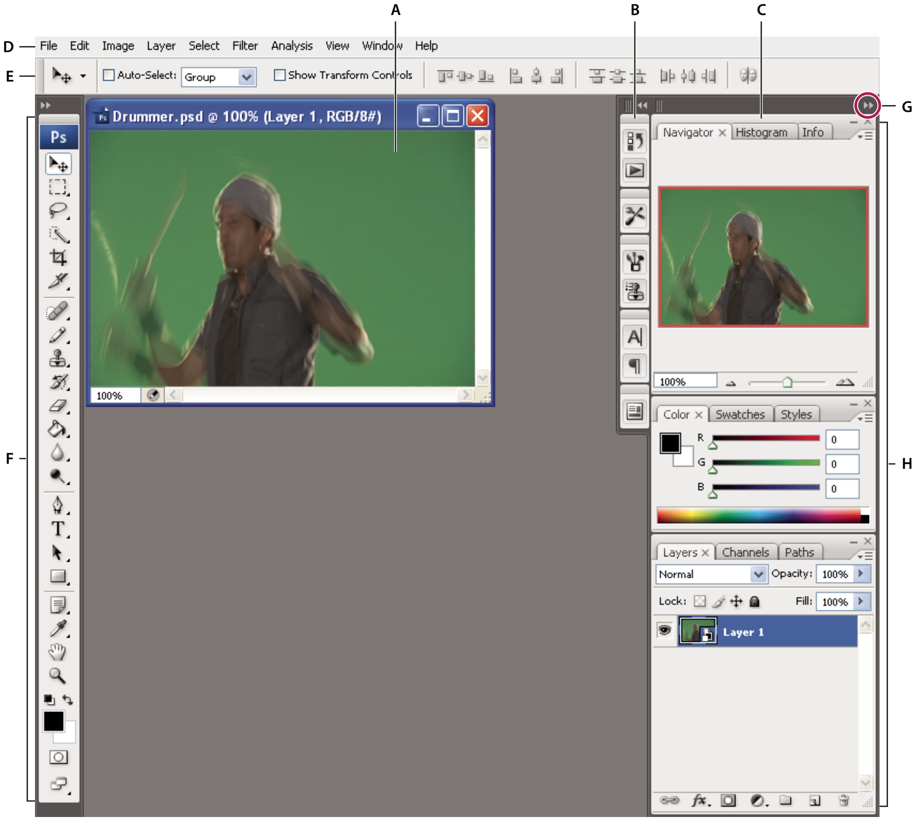
Workspace overview

You create and manipulate your documents and files using various elements such as panels, bars, and windows. Any arrangement of these elements is called a *workspace*. When you first start an Adobe Creative Suite component, you see the default workspace, which you can customize for the tasks you perform there. For instance, you can create one workspace for editing and another for viewing, save them, and switch between them as you work.

You can restore the default workspace at any time by choosing the default option on the Window > Workspace menu.

Although default workspaces vary across Flash, Illustrator, InCopy, InDesign, and Photoshop, you manipulate the elements much the same way in all of them. The Photoshop default workspace is typical:

- The *menu bar* across the top organizes commands under menus.
- The *Tools panel* (called the *Tools palette* in Photoshop) contains tools for creating and editing images, artwork, page elements, and so on. Related tools are grouped together.
- The *Control panel* (called the *options bar* in Photoshop) displays options for the currently selected tool. (Flash has no Control panel.)
- The *Document window* (called the *Stage* in Flash) displays the file you're working on.
- *Panels* (called *palettes* in Photoshop) help you monitor and modify your work. Examples include the Timeline in Flash and the Layers palette in Photoshop. Certain panels are displayed by default, but you can add any panel by selecting it from the Window menu. Many panels have menus with panel-specific options. Panels can be grouped, stacked, or docked.



Default Photoshop workspace
***A.** Document window **B.** Dock of panels collapsed to icons **C.** Panel title bar **D.** Menu bar **E.** Options bar **F.** Tools palette **G.** Collapse To Icons button **H.** Three palette (panel) groups in vertical dock*

For a video on understanding the workspace, see www.adobe.com/go/vid0187.

Hide or show all panels

- (Illustrator, InCopy, InDesign, Photoshop) To hide or show all panels, including the Tools panel and options bar or Control panel, press Tab.
- (Illustrator, InCopy, InDesign, Photoshop) To hide or show all panels except the Tools panel and options bar or Control panel, press Shift+Tab.

You can temporarily display panels hidden by these techniques by moving the pointer to the edge of the application window (Windows) or to the edge of the monitor (Mac OS) and hovering over the strip that appears.

- (Flash) To hide or show all panels, press F4.

Display panel menu options

❖ Position the pointer on the panel menu icon ▾☰ in the upper-right corner of the panel, and press the mouse button.

(Illustrator) Adjust panel brightness

❖ In User Interface preferences, move the Brightness slider. This control affects all panels, including the Control panel.

Reconfigure the Tools panel

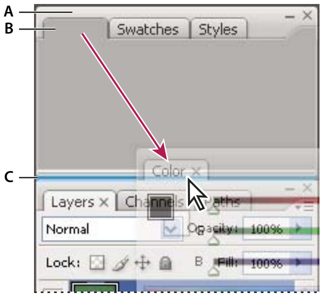
You can display the tools in the Tools panel in a single column, or side by side in two columns.

In InDesign, you also can switch from single-column to double-column display by setting an option in Interface preferences.

❖ Click the double arrow at the top of the Tools panel.

Customize the workspace

To create a custom workspace, move and manipulate panels (called *palettes* in Photoshop and in Adobe Creative Suite 2 components).



Narrow blue drop zone indicates Color panel will be docked on its own above Layers panel group.
***A.** Title bar **B.** Tab **C.** Drop zone*

You can save custom workspaces and switch among them.

In Photoshop, you can change the font size of the text in the options bar, palettes, and tool tips. Choose a size from the UI Font Size menu in General preferences.

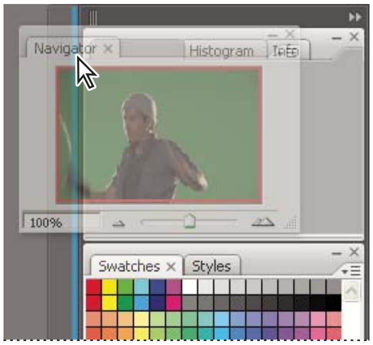
***Note:** For a video on customizing the workspace in Illustrator, see www.adobe.com/go/vid0032. For a video on customizing the workspace in InDesign, see www.adobe.com/go/vid0065.*

Dock and undock panels

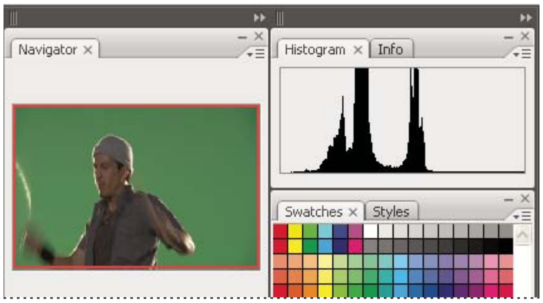
A *dock* is a collection of panels or panel groups displayed together, generally in a vertical orientation. You dock and undock panels by moving them into and out of a dock.

***Note:** Docking is not the same as stacking. A stack is a collection of free-floating panels or panel groups, joined top to bottom.*

- To dock a panel, drag it by its tab into the dock, at the top, bottom, or in between other panels.
- To dock a panel group, drag it by its title bar (the solid empty bar above the tabs) into the dock.
- To remove a panel or panel group, drag it out of the dock by its tab or title bar. You can drag it into another dock or make it free-floating.



Navigator panel being dragged out to new dock, indicated by blue vertical highlight



Navigator panel now in its own dock

To prevent panels from filling all space in a dock, drag the bottom edge of the dock up so it no longer meets the edge of the workspace.

Move panels

As you move panels, you see blue highlighted *drop zones*, areas where you can move the panel. For example, you can move a panel up or down in a dock by dragging it to the narrow blue drop zone above or below another panel. If you drag to an area that is not a drop zone, the panel floats freely in the workspace.

- To move a panel, drag it by its tab.
- To move a panel group or a stack of free-floating panels, drag the title bar.

Press Ctrl (Windows) or Control (Mac OS) while moving a panel to prevent it from docking.

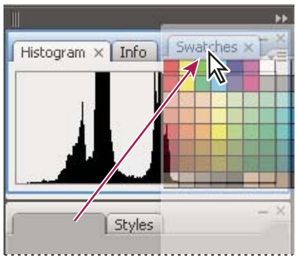
Add and remove docks and panels

If you remove all panels from a dock, the dock disappears. You can create new docks by moving panels to drop zones next to existing docks or at the edges of the workspace.

- To remove a panel, click its close icon (the X at the upper-right corner of the tab), or deselect it from the Window menu.
- To add a panel, select it from the Window menu and dock it wherever you wish.

Manipulate panel groups

- To move a panel into a group, drag the panel's tab to the highlighted drop zone at the top of the group.

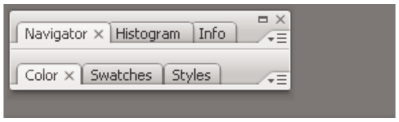


Adding a panel to a panel group

- To rearrange panels in a group, drag a panel’s tab to a new location in the group.
- To remove a panel from a group so that it floats freely, drag the panel by its tab outside the group.
- To make a panel appear at the front of its group, click its tab.
- To move grouped panels together, drag their title bar (above the tabs).

Stack free-floating panels

When you drag a panel out of its dock but not into a drop zone, the panel floats freely, allowing you to position it anywhere in the workspace. Panels may also float in the workspace when first selected from the Window menu. You can stack free-floating panels or panel groups together so that they move as a unit when you drag the topmost title bar. (Panels that are part of a dock cannot be stacked or moved as a unit in this way.)



Free-floating stacked panels

- To stack free-floating panels, drag a panel by its tab to the drop zone at the bottom of another panel.
- To change the stacking order, drag a panel up or down by its tab.

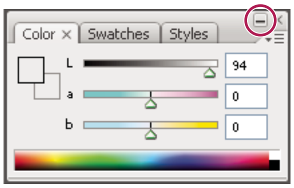
Note: *Be sure to release the tab over the narrow drop zone between panels, rather than the broad drop zone in a title bar.*

- To remove a panel or panel group from the stack, so that it floats by itself, drag it out by its tab or title bar.

Resize or minimize panels

- To resize a panel, drag any side of the panel or drag the size box at its lower-right corner. Some panels, such as the Color panel in Photoshop, cannot be resized by dragging.
- To change the width of all the panels in a dock, drag the gripper at the top left of the dock.
- To minimize a panel, panel group, or stack of panels, click the Minimize button in its title bar.

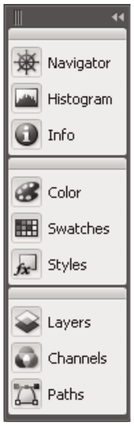
You can open a panel menu even when the panel is minimized.



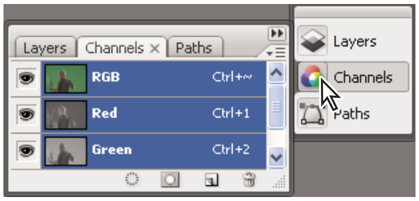
Minimize button

Manipulate panels collapsed to icons

Collapse panels to icons to reduce clutter on the workspace. (In some cases, panels are collapsed to icons in the default workspace.) Click a panel icon to expand the panel. You can expand only one panel or panel group at a time.



Panels collapsed to icons



Panels expanded from icons

- To collapse or expand all panels in a dock, click the double arrow at the top of the dock.
- To resize panel icons so that you see only the icons (and not the labels), drag the gripper ▥ at the top of the dock toward the icons until the text disappears. (To display the icon text again, drag the gripper away from the panels.)
- To expand a single panel icon, click it.
- To collapse an expanded panel back to its icon, click its tab, its icon, or the double arrow in the panel's title bar.

If you select Auto-Collapse Icon Panels from the Interface or User Interface Options preferences, an expanded panel icon will collapse automatically when you click away from it.

- To add a panel or panel group to an icon dock, drag it in by its tab or title bar. (Panels are automatically collapsed to icons when added to an icon dock.)
- To move a panel icon (or panel icon group), drag the bar that appears above the icon. You can drag panel icons up and down in the dock, into other docks (where they appear in the panel style of that dock), or outside the dock (where they appear as free-floating, expanded panels).

Save, delete, and switch between workspaces

By saving the current size and position of panels as a named workspace, you can restore that workspace even if you move or close a panel. The names of saved workspaces appear in the Window > Workspace menu.

In Photoshop, the saved workspace can include a specific keyboard shortcut set and menu set.

Save a custom workspace

1 With the workspace in the configuration you want to save, do one of the following:

- (Photoshop, Illustrator, InDesign) Choose Window > Workspace > Save Workspace.
- (Flash) Choose Window > Workspace > Save Current, or choose Save Current from the Workspace menu in the Edit bar.
- (Photoshop) Choose Save Workspace from the Workspace menu in the options bar.

2 Type a name for the workspace.

3 (Photoshop) Under Capture, select one or more options:

Palette Locations Saves the current palette locations.

Keyboard Shortcuts Saves the current set of keyboard shortcuts.

Menus Saves the current set of menus.

4 Click OK.

Display or switch between workspaces

Flash, Illustrator, InDesign, and Photoshop include preset workspaces designed to make certain tasks easier.

- Choose Window > Workspace, and select a workspace.
- (Photoshop) Select a workspace from the Workspace menu in the options bar.
- (Flash) Select a workspace from the Workspace menu in the Edit bar.

💡 *(InDesign and Photoshop) Assign keyboard shortcuts to each workspace to navigate among them quickly.*

Delete a custom workspace

- (Illustrator) Choose Window > Workspace > Manage Workspaces, select the workspace, and then click the Delete icon.
- (InDesign) Choose Window > Workspace > Delete Workspace, select the workspace, and then click Delete.
- (Flash) Choose Manage from the Workspace menu in the Edit bar, select the workspace, and then click Delete. Alternatively, choose Window > Workspace > Manage, select the workspace, and then click Delete.
- (Photoshop) Choose Delete Workspace from the Workspace menu in the options bar. Alternatively, choose Window > Workspace > Delete Workspace, select the workspace, and then click Delete.

(Photoshop) Start with the last or default palette locations

When you start Photoshop, palettes can either appear in their original default locations, or appear as you last used them.

❖ In Interface preferences:

- To display palettes in their last locations on startup, select Remember Palette Locations.
- To display palettes in their default locations on startup, deselect Remember Palette Locations.

Using the Stage and Tools panel

Welcome screen overview

When Flash is running with no documents open, the Welcome screen appears. The Welcome screen contains the following four areas:

Open a Recent Item Lets you open your most recent documents (click the Open icon).

Create New Lists Flash file types, such as Flash documents and ActionScript™ files.

Create from Template Lists the templates most commonly used to create Flash documents.

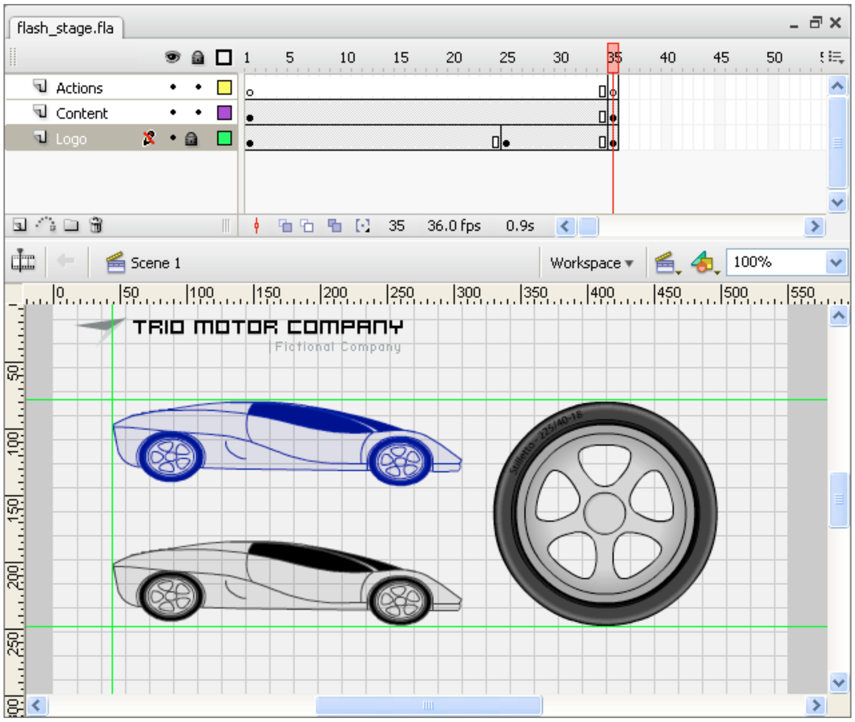
Extend Links to the Flash Exchange website, where you can download helper applications, extensions, and related information.

The Welcome screen also offers quick access to Help resources. You can take a tour of Flash, learn about documentation resources, and find Adobe Authorized Training facilities.

- To hide the Welcome screen, select Don't Show Again.
- To show the Welcome screen, select Edit > Preferences (Windows) or select Flash > Preferences (Macintosh), and select Show Welcome screen in the General category.

Using the Stage

The Stage is the rectangular area where you place graphic content when creating Flash documents. The Stage in the authoring environment represents the rectangular space in Flash Player or in a web browser window where your document appears during playback. To change the view of the Stage as you work, zoom in and out. To help you position items on the Stage, you can use the grid, guides, and rulers.



The Timeline and Stage with content

For a video tutorial about the Flash interface, see www.adobe.com/go/vid0116.

Zoom the Stage

To view the entire Stage on the screen, or to view a particular area of your drawing at high magnification, change the magnification level. The maximum magnification depends on the resolution of your monitor and the document size. The minimum value for zooming out on the Stage is 8%. The maximum value for zooming in on the Stage is 2000%.

- To zoom in on an element, select the Zoom tool in the Tools panel, and click the element. To switch the Zoom tool between zooming in or out, use the Enlarge or Reduce modifiers (in the options area of the Tools panel when the Zoom tool is selected) or Alt-click (Windows) or Option-click (Macintosh).
- To zoom in so that a specific area of your drawing fills the window, drag a rectangular selection on the Stage with the Zoom tool.
- To zoom in on or out of the entire Stage, select View > Zoom In or View > Zoom Out.
- To zoom in or out by a specified percentage, select View > Magnification, and select a percentage from the submenu or select a percentage from the Zoom control at the upper-right corner of the Timeline.
- To scale the Stage so that it fits completely in the application window, select View > Magnification > Fit in Window.
- To show the contents of the current frame, select View > Magnification > Show All, or select Show All from the Zoom control at the upper-right side of the application window. If the scene is empty, the entire Stage appears.
- To show the entire Stage, select View > Magnification > Show Frame or select Show Frame from the Zoom control at the upper-right corner of the Timeline.
- To show the workspace surrounding the Stage, or to view elements in a scene that are partly or completely outside of the Stage area, select View > Pasteboard. The pasteboard appears in light gray. For example, to have a bird fly into a frame, initially position the bird outside of the Stage in the pasteboard and animate it into the Stage area.

Move the view of the Stage

When the Stage is magnified, you may not be able to see all of it. To change the view without having to change the magnification, use the Hand tool to move the Stage.

❖ In the Tools panel, select the Hand tool and drag the Stage. To temporarily switch between another tool and the Hand tool, hold down the Spacebar and click the tool in the Tools panel.

Use rulers

When rulers show, they appear along the top and left sides of the document. You can change the unit of measure used in the rulers from the default of pixels to another unit. When you move an element on the Stage with the rulers displayed, lines indicating the element's dimensions appear on the rulers.

- To show or hide rulers, select View > Rulers.
- To specify the rulers' unit of measure for a document, select Modify > Document, and select a unit from the Ruler Units menu.

See also

"Snapping" on page 180

Use guides

When rulers show (View > Rulers), you can drag horizontal and vertical guides from the rulers onto the Stage.

When you create nested timelines, draggable guides appear on the Stage only when the Timeline in which they were created is active.

To create custom guides or irregular guides, use guide layers.

- To display or hide the drawing guides, select View > Guides > Show Guides.

***Note:** If the grid is visible and Snap to Grid is turned on when you create guides, guides snap to the grid.*

- To turn snapping to guides on or off, select View > Snapping > Snap to Guides.

***Note:** Snapping to guides takes precedence over snapping to the grid in places where guides fall between grid lines.*

- To move a guide, click anywhere on the ruler with the Selection tool and drag the guide to the desired place on the Stage.
- To remove a guide, use the Selection tool with guides unlocked to drag the guide to the horizontal or vertical ruler.
- To lock guides, select View > Guides > Lock Guides or use the Lock Guides option in the Edit Guides (View > Guides > Edit Guides) dialog box.
- To clear guides, select View > Guides > Clear Guides. If you are in document-editing mode, all guides in the document are cleared. If you are in symbol-editing mode, only guides used in symbols are cleared.

See also

"Use guide layers" on page 39

Set guide preferences

1 Select View > Guides > Edit Guides and do any of the following:

- To set Color, click the triangle in the color box and select a guide line color from the palette. The default guide color is green.
- To display or hide guides, select or deselect Show Guides.
- To turn snapping to guides on or off, select or deselect Snap To Guides.
- Select or deselect Lock Guides.
- To set Snap Accuracy, select an option from the pop-up menu.
- To remove all guides, click Clear All. Clear All removes all guides from the current scene.
- To save the current settings as the default, click Save Default.

2 Click OK.

Use the grid

The grid appears in a document as a set of lines behind the artwork in all scenes.

Display or hide the drawing grid

❖ Do one of the following:

- Select View > Grid > Show Grid.
- Press Control+" (quote) (Windows) or Command+" (quote) (Macintosh).

Turn snapping to grid lines on or off

❖ Select View > Snapping > Snap to Grid.

Set grid preferences

1 Select View > Grid > Edit Grid and select from the options.

2 To save the current settings as the default, click Save Default.

About the main toolbar and edit bar

The menu bar at the top of the application window contains menus with commands for controlling functionality.

The edit bar, at the top of the Stage, contains controls and information for editing scenes and symbols, and for changing the magnification level of the Stage.

See also

"Using symbols, instances, and library assets" on page 207

"Working with scenes" on page 74

Tools panel overview

The tools in the Tools panel let you draw, paint, select, and modify artwork, as well as change the view of the Stage. The Tools panel is divided into four sections:

- The tools area contains drawing, painting, and selection tools.
- The view area contains tools for zooming and panning in the application window.
- The colors area contains modifiers for stroke and fill colors.
- The options area contains modifiers for the currently selected tool. Modifiers affect the tool's painting or editing operations.

To specify which tools to display in the authoring environment, use the Customize Tools Panel dialog box.

See also

"Using Flash drawing and painting tools" on page 164

"Selecting objects" on page 196

Use the Tools panel

To show or hide the Tools panel, select Window > Tools.

Select tools

❖ Do one of the following:

- Click the tool in the Tools panel. Depending on the tool you select, a set of modifiers might appear in the options area at the bottom of the Tools panel.
- Press the tool's keyboard shortcut. To view the keyboard shortcuts, select Edit > Keyboard Shortcuts (Windows) or Flash > Keyboard Shortcuts (Macintosh). On the Macintosh, you might need to move the mouse to see the new pointer appear.

- To select a tool located in the pop-up menu for a visible tool such as the Rectangle tool, press the icon of the visible tool and select another tool from the pop-up menu.

Customize the Tools panel

To specify which tools appear in the authoring environment, use the Customize Tools Panel dialog box to add or remove tools from the Tools panel.

When more than one tool appears in a location, the top tool in the group (the most recently used) appears with an arrow in the lower-right corner of its icon. This arrow indicates that additional tools are present in a pop-up menu. The same keyboard shortcut functions for all tools in the pop-up menu. When you press and hold the mouse button on the icon, the other tools in the group appear in a pop-up menu.

1 To show the Customize Tools Panel dialog box, do one of the following:

- (Windows) Select Edit > Customize Tools panel.
- (Macintosh) Select Flash > Customize Tools panel.

The Available Tools menu indicates the tools that are currently available. The Current Selection menu indicates the tools currently assigned to the selected location in the Tools panel.

2 To browse through the tools to specify the location to assign to another tool, click a tool in the Tools panel image or use the arrows.

3 To add a tool to the selected location, select the tool in the Available Tools list and click Add. You can assign a tool to more than one location.

4 To remove a tool from the selected location, select the tool in the Current Selection scroll list and click Remove.

5 To restore the default Tools Panel layout, click Restore Default in the Customize Tools Panel dialog box.

6 Click OK to apply your changes and close the Customize Tools Panel dialog box.

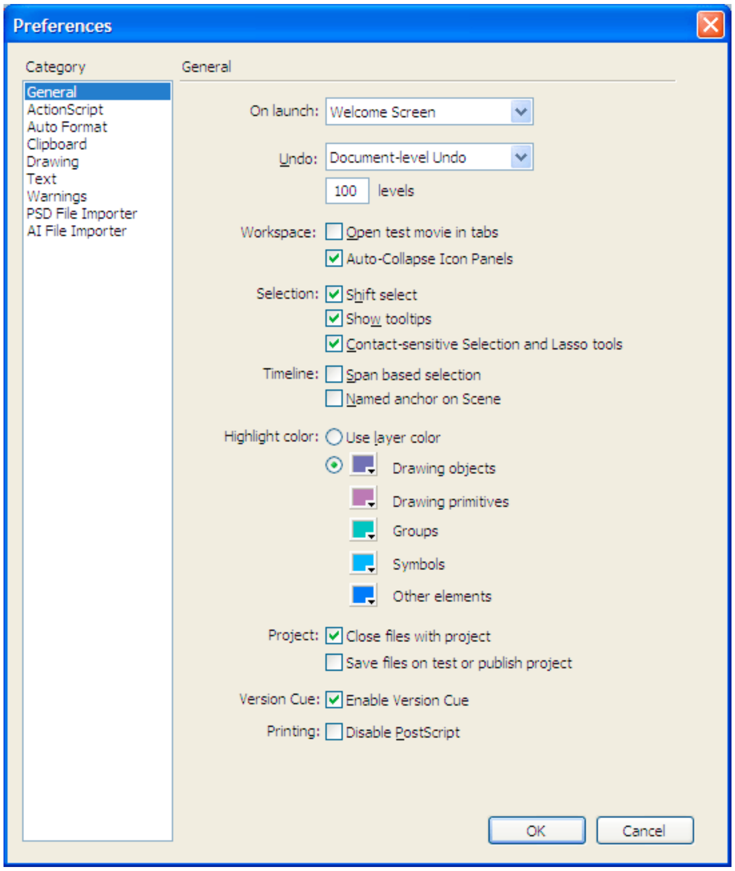
Use context menus

Context menus contain commands relevant to the current selection. For example, when you select a frame in the Timeline window, the context menu contains commands for creating, deleting, and modifying frames and keyframes. Context menus exist for many items and controls in many locations, including on the Stage, in the Timeline, in the Library panel, and in the Actions panel.

❖ Right-click (Windows) or Control-click (Macintosh) an item.

Set preferences in Flash

You can set preferences for general application operations, editing operations, and clipboard operations.



Preferences

Category

General
ActionScript
Auto Format
Clipboard
Drawing
Text
Warnings
PSD File Importer
AI File Importer

General

On launch: Welcome Screen

Undo: Document-level Undo

100 levels

Workspace: □ Open test movie in tabs
☑ Auto-Collapse Icon Panels

Selection: ☑ Shift select
☑ Show tooltips
☑ Contact-sensitive Selection and Lasso tools

Timeline: □ Span based selection
□ Named anchor on Scene

Highlight color: ○ Use layer color
● Drawing objects
Drawing primitives
Groups
Symbols
Other elements

Project: ☑ Close files with project
□ Save files on test or publish project

Version Cue: ☑ Enable Version Cue

Printing: □ Disable PostScript

The General category in the Preferences dialog box

See also

"Specify drawing preferences" on page 163

"Change the display of frames in the Timeline" on page 35

"About the Timeline" on page 33

"Creating and managing documents" on page 51

"Substituting missing fonts" on page 263

"Set Pen tool preferences" on page 173

"AI File Importer preferences" on page 143

"PSD file import preferences" on page 150

Set preferences

1 Select Edit > Preferences (Windows) or Flash > Preferences (Macintosh).

2 Make a selection in the Category list and select from the respective options.

Set AutoFormat preferences for ActionScript

❖ Select any of the options. To see the effect of each selection, look in the Preview pane.

Set text preferences

- For Font Mapping Default, select a font to use when substituting missing fonts in documents you open in Flash.
- For Vertical Text options, select Default Text Orientation (deselected by default).
- To reverse the default text display direction, select Right To Left Text Flow (deselected by default).
- To turn off kerning for vertical text, select No Kerning (deselected by default). Turning off kerning is useful to improve spacing for some fonts that use kerning tables.
- For Input Method, select the appropriate language.

Set warning preferences

- To receive a warning when you try to save documents with content that is specific to the Adobe® Flash® CS3 Professional authoring tool as a Flash 8 file, select Warn On Save For Adobe Flash 8 Compatibility (default).
- To receive a warning when you open a Flash document that uses fonts that are not installed on your computer, select Warn On Missing Fonts (default).
- To receive a warning if the URL for a document changed since the last time you opened and edited it, select Warn On URL Changes In Launch And Edit.
- To place a red X over any Generator objects as a reminder that Generator objects are not supported in Flash 8, select Warn On Reading Generator Content.
- To receive an alert when Flash inserts frames in your document to accommodate audio or video files that you import, select Warn On Inserting Frames When Importing Content.
- To receive an alert when selecting Default Encoding could potentially lead to data loss or character corruption, select Warn On Encoding Conflicts When Exporting .as Files. (For example, if you create a file with English, Japanese, and Korean characters and select Default Encoding on an English system, the Japanese and Korean characters are corrupted.)
- To receive a warning when you attempt to edit a symbol with timeline effects applied to it, select Warn On Conversion Of Effect Graphic Objects.
- To receive a warning when you export a document to this earlier version of Flash Player, select Warn On Exporting To Flash Player 6 r65.
- To receive a warning when you create a site in which the local root folder overlaps with another site, select Warn On Sites With Overlapped Root Folder.
- To receive a warning when you convert a symbol with a behavior attached to a symbol of a different type—for example, when you convert a movie clip to a button—select Warn On Behavior Symbol Conversion.
- To receive a warning when you convert a symbol to a symbol of a different type, select Warn On Symbol Conversion.
- To receive a warning when Flash converts a graphic object drawn in Object Drawing mode to a group, select Warn On Automatically Converting From Drawing Object To Group.
- To display warnings on controls for features not supported by the Flash Player version that the current FLA file is targeting in its Publish settings, select Show Incompatibility Warnings On Feature Controls.

Set General preferences

On Launch Specify which document opens when you start the application.

Undo To set the number of undo or redo levels, enter a value from 2 to 300. Undo levels require memory; the more undo levels you use, the more system memory is consumed. The default is 100.

Document- or Object-level undo Document-level undo maintains a single list of all your actions for the entire Flash document. Object-level undo maintains separate lists of your actions for each object in your document. Object-level lets you undo an action on one object without having to also undo actions on other objects that might have been modified more recently than the target object.

Printing (Windows only) To disable PostScript output when printing to a PostScript printer, select Disable PostScript. By default, this option is deselected. Select this option if you have problems printing to a PostScript printer; however, this option slows down printing.

Test Movie To open a new document tab in the application window when you select Control > Test Movie, select Open Test Movie In Tabs. The default is to open the test movie in its own window.

Selection To control how multiple elements are selected, select or deselect Shift Select. When Shift Select is off, clicking additional elements adds them to the current selection. When Shift Select is on, clicking additional elements deselects other elements unless you hold down Shift.

Show Tooltips Shows tooltips when the pointer pauses over a control. To hide the tooltips, deselect this option.

Contact Sensitive Select objects when any part of them is included in the marquee rectangle when dragging with the Selection or Lasso tools. The default is that objects are only selected when the tool's marquee rectangle completely surrounds the object.

Timeline To use span-based selection in the Timeline, rather than the default frame-based selection, select Span Based Selection.

Named Anchor On Scene Make the first frame of each scene in a document a named anchor. Named anchors let you use the Forward and Back buttons in a browser to jump from scene to scene.

Highlight Color To use the current layer's outline color, select a color from the panel, or select Use Layer Color.

Project To have all files in a project close when the project file is closed, select Close Files With Project.

Save Files On Test Or Publish Project Save each file in a project whenever the project is tested or published.

Clipboard preferences

Bitmaps (Windows only)

To specify Color Depth and Resolution parameters for bitmaps copied to the clipboard, select their respective options.

To apply anti-aliasing, select Smooth.

To specify the amount of RAM that is used when placing a bitmap image on the Clipboard, enter a value in the Size Limit text field. Increase this value when working with large or high-resolution bitmap images.

Gradient Quality (Windows only) To specify the quality of gradient fills placed in the Windows metafile, select an option. Choosing a higher quality increases the time required to copy artwork. To specify gradient quality when pasting items to a location outside of Flash, use this setting. When you are pasting in Flash, the full gradient quality of the copied data is preserved regardless of the Gradients setting on the Clipboard.

PICT Settings (Macintosh only) •Type To preserve data copied to the Clipboard as vector artwork, select Objects. To convert the copied artwork to a bitmap, select one of the bitmap formats.

- **Resolution** Enter a value.

- **Include PostScript** Select to include PostScript data.

- **Gradients** To specify gradient quality in the PICT file, select an option. Choosing a higher quality increases the time required to copy artwork. To specify gradient quality when pasting items to a location outside of Flash, use the

Gradients setting. When you are pasting in Flash, the full gradient quality of the copied data is preserved regardless of the Gradient setting.

- **FreeHand Text** To keep text editable in a pasted FreeHand file, select Maintain Text As Blocks.

Customize keyboard shortcuts

To match the shortcuts you use in other applications, or to streamline your workflow, select keyboard shortcuts. By default, Flash uses built-in keyboard shortcuts designed for the application. You can also select a built-in keyboard shortcut set from one of several graphics applications.

View or print the current set of keyboard shortcuts

1 Select Edit > Keyboard Shortcuts (Windows) or Flash > Keyboard Shortcuts (Macintosh).

2 In the Keyboard Shortcuts dialog box, select the shortcut set to view from the Current Set pop-up menu.

3 Click the Export Set As HTML button .

4 Select a name and location for the exported HTML file. The default file name is the name of the selected shortcut set.

5 Click Save.

6 Find the exported file in the folder you selected and open the file in a web browser.

7 To print the file, use the browser's Print command.

Select a keyboard shortcut set

1 Select Edit > Keyboard Shortcuts (Windows) or Flash > Keyboard Shortcuts (Macintosh).

2 In the Keyboard Shortcuts dialog box, select a shortcut set from the Current Set pop-up menu.

Create a keyboard shortcut set

1 Select a keyboard shortcut set and click the Duplicate Set button.

2 Enter a name for the new shortcut set and click OK.

Rename a custom keyboard shortcut set

1 In the Keyboard Shortcuts dialog box, select a shortcut set from the Current Set pop-up menu.

2 Click the Rename Set button, enter a new name, and click OK.

Add or remove a keyboard shortcut

1 Select Edit > Keyboard Shortcuts (Windows) or Flash > Keyboard Shortcuts (Macintosh) and select the set to modify.

2 From the Commands pop-up menu, select a category to view shortcuts for the selected category.

3 In the Commands list, select the command for which you want to add or remove a shortcut. An explanation of the selected command appears in the description area in the dialog box.

4 Do one of the following:

- To add a shortcut, click the Add Shortcut (+) button.
- To remove a shortcut, click the Remove Shortcut (-) button and proceed to step 6.

5 If you are adding a shortcut, enter the new shortcut key combination in the Press Key box.

***Note:** To enter the key combination, press the keys on the keyboard. You do not need to spell out key names, such as Control, Option, and so on.*

6 Click Change.

7 Repeat this procedure to add or remove additional shortcuts, and click OK.

Delete a keyboard shortcut set

1 Select Edit > Keyboard Shortcuts (Windows) or Flash > Keyboard Shortcuts (Macintosh). In the Keyboard Shortcuts dialog box, click Delete Set.

2 In the Delete Set dialog box, select a shortcut set and click Delete.

***Note:** You cannot delete the keyboard shortcut sets built into Flash.*

Create custom keyboard shortcuts

You can create and modify keyboard shortcuts.

Customize keyboard shortcuts

1 Select Edit > Keyboard Shortcuts (Windows) or Flash > Keyboard Shortcuts (Macintosh).

The Keyboard Shortcuts dialog box appears.

2 Use the following options to add, delete, or edit keyboard shortcuts:

Current Set Lets you choose a set of predetermined shortcuts (listed at the top of the menu), or any custom set you've defined.

Commands Lets you select a category of commands to edit (for example, menu commands). The command list displays the commands associated with the category you selected from the Commands pop-up menu, along with the assigned shortcuts. The Menu Commands category displays this list as a tree view that replicates the structure of the menus. The other categories list the commands by name (such as Quit Application), in a flat list.

Shortcuts Displays all shortcuts assigned to the selected command.

Add Item ✚ Adds a new shortcut to the current command. To add a new blank line to the Shortcuts box, click this button. To add a new keyboard shortcut for this command, enter a new key combination and click Change. Each command can have two different keyboard shortcuts; if two shortcuts are already assigned to a command, the Add Item button does nothing.

Remove Item ▬ Removes the selected shortcut from the list of shortcuts.

Press Key Displays the key combination you enter when you're adding or changing a shortcut.

Change Adds the key combination shown in the Press Key box to the list of shortcuts, or changes the selected shortcut to the specified key combination.

Duplicate Duplicates the current set. Give the new set a name; the default name is the current set's name with the word *copy* appended to it.

Rename Renames the current set.

Export Set As HTML Saves the current set in an HTML table format for easy viewing and printing. Open the HTML file in your browser and print the shortcuts for easy reference.

Delete Deletes a set. You cannot delete the active set.

3 Click OK.

Remove a shortcut from a command

1 From the Commands pop-up menu, select a command category, select a command from the Commands list, and select a shortcut.

2 Click Remove Item (-).

Add a shortcut to a command

1 From the Commands pop-up menu, select a command category and select a command.

2 Prepare to add a shortcut by doing one of the following:

- If fewer than two shortcuts are already assigned to the command, click Add Item ✚. A new blank line appears in the Shortcuts box, and the insertion point moves to the Press Key box.
- If two shortcuts are already assigned to the command, select one of them to be replaced by the new shortcut, and click in the Press Key box.

3 Press a key combination.

***Note:** If a problem occurs with the key combination (for example, if the key combination is already assigned to another command), an explanatory message appears just below the Shortcuts box and you may be unable to add or edit the shortcut.*

4 Click Change.

Edit an existing shortcut

1 From the Commands pop-up menu, select a command category, select a command from the Commands list, and select a shortcut to change.

2 Click in the Press Key box, enter a new key combination, and click Change.

***Note:** If a problem occurs with the key combination (for example, if the key combination is already assigned to another command), an explanatory message appears just below the Shortcuts box and you may be unable to add or edit the shortcut.*

The Timeline

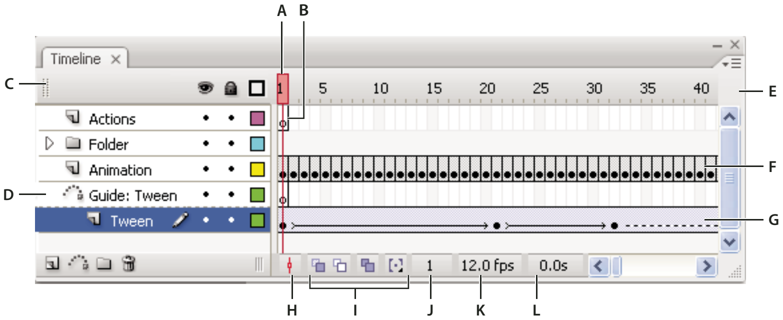
About the Timeline

The 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. As a document plays, the playhead moves from left to right through the Timeline.

The Timeline status displayed at the bottom of the Timeline indicates the selected frame number, the current frame rate, and the elapsed time to the current frame.

***Note:** When an animation is played, the actual frame rate is displayed; this may differ from the document’s frame rate setting if the computer can’t calculate and display the animation quickly enough.*



Parts of the Timeline
A. *Playhead* ***B.*** *Empty keyframe* ***C.*** *Timeline header* ***D.*** *Guide layer icon* ***E.*** *Frame View pop-up menu* ***F.*** *Frame-by-frame animation* ***G.*** *Tweened animation* ***H.*** *Scroll To Playhead button* ***I.*** *Onion-skinning buttons* ***J.*** *Current Frame indicator* ***K.*** *Frame Rate indicator* ***L.*** *Elapsed Time indicator*

The Timeline shows where animation occurs in a document, including frame-by-frame animation, tweened animation, and motion paths.

Controls in the layers section of the Timeline let you hide, show, lock, or unlock layers, as well as display layer contents as outlines. You can drag frames to a new location on the same layer or to a different layer.

For a video tutorial about the Timeline, keyframes, and frame rates, see www.adobe.com/go/vid0123.

See also

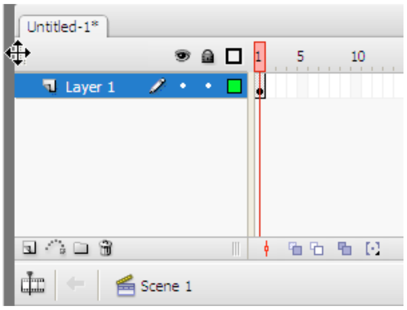
“Manage frames and keyframes in the Timeline” on page 69

“Creating motion” on page 228

Change the appearance of the Timeline

By default, the Timeline appears at the top of the main application window, above the Stage. To change its position, detach the Timeline from the Stage and float it in its own window or dock it to any other panel you choose. You can also hide the Timeline.

To change the number of layers and frames that are visible, resize the Timeline. To view additional layers when the Timeline contains more layers than can be displayed, use the scroll bars on the right side of the Timeline.



Dragging the Timeline

- To move the Timeline when it is docked to the application window, drag the gripper (2 dotted vertical bars) at the upper-left corner of the Timeline.

- To dock an undocked Timeline to the application window, drag the gripper (2 dotted vertical bars) to the top of the application window.
- To dock an undocked Timeline to other panels, drag the Timeline title bar tab to the location you choose. To prevent the Timeline from docking to other panels, press Control while you drag. A blue bar appears to indicate where the Timeline will dock.
- To lengthen or shorten layer name fields in the Timeline panel, drag the bar separating the layer names and the frames portions of the Timeline.

Resize the Timeline

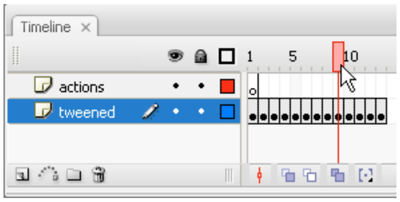
- If the Timeline is docked to the main application window, drag the bar separating the Timeline from the Stage area.
- If the Timeline is not docked to the main application window, drag the lower-right corner (Windows) or the size box in the lower-right corner (Macintosh).

Move the playhead

The playhead moves through the timeline as a document plays to indicate the current frame displayed on the Stage. The Timeline header shows the frame numbers of the animation. To display a frame on the Stage, move the playhead to the frame in the Timeline.

To display a specific frame when you're working with a large number of frames that can't all be displayed in the Timeline at once, move the playhead along the Timeline.

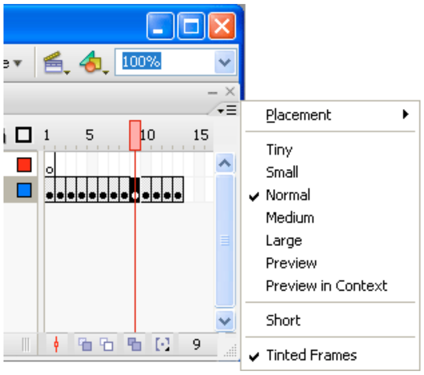
- To go to a frame, click the frame's location in the Timeline header, or drag the playhead to the desired position.
- To center the Timeline on the current frame, click the Scroll To Playhead button at the bottom of the Timeline.



Moving the playhead

Change the display of frames in the Timeline

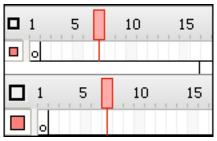
1 To display the Frame View pop-up menu, click Frame View in the upper-right corner of the Timeline.



Frame View pop-up menu

2 Select from the following options:

- To change the width of frame cells, select Tiny, Small, Normal, Medium, or Large. (The Large frame-width setting is useful for viewing the details of sound waveforms.)
- To decrease the height of frame cell rows, select Short.



Short and Normal frame view options

- To turn the tinting of frame sequences on or off, select Tinted Frames.
- To display thumbnails of the content of each frame scaled to fit the Timeline frames, select Preview. This can cause the apparent content size to vary and requires extra screen space.
- To display thumbnails of each full frame (including empty space), select Preview In Context. This is useful for viewing the way elements move in their frames over the course of the animation, but previews are generally smaller than with the Preview option.

About layers

Layers help you organize the artwork in your document. You can draw and edit objects on one layer without affecting objects on another layer. In areas of the Stage with nothing on a layer, you can see through it to the layers below.

To draw, paint, or otherwise modify a layer or folder, select the layer in the Timeline to make it active. A pencil icon next to a layer or folder name in the Timeline indicates that the layer or folder is active. Only one layer can be active at a time (although more than one layer can be selected at a time).

When you create a Flash document, it contains only one layer. To organize the artwork, animation, and other elements in your document, add more layers. You can also hide, lock, or rearrange layers. The number of layers you can create is limited only by your computer's memory, and layers do not increase the file size of your published SWF file. Only the objects you place into layers add to the file size.

To organize and manage layers, create layer folders and place layers in them. You can expand or collapse layer folders in the Timeline without affecting what you see on the Stage. Use separate layers or folders for sound files, ActionScript, frame labels, and frame comments. This helps you find these items quickly to edit them.

To help create sophisticated effects, use special guide layers to make drawing and editing easier, and mask layers.

Create layers and layer folders

When you create a layer or folder, it appears above the selected layer. The newly added layer becomes the active layer.

Create a layer

❖ Do one of the following:

- Click the Insert Layer button at the bottom of the Timeline.
- Select Insert > Timeline > Layer.
- Right-click (Windows) or Control-click (Macintosh) a layer name in the Timeline and select Insert Layer from the context menu.

Create a layer folder

❖ Do one of the following:

- Select a layer or folder in the Timeline and select Insert > Timeline > Layer Folder.
- Right-click (Windows) or Control-click (Macintosh) a layer name in the Timeline and select Insert Folder from the context menu. The new folder appears above the layer or folder you selected.

View layers and layer folders

A red X next to the name of a layer or folder in the Timeline indicates that a layer or folder is hidden. In the Publish Settings, you can choose whether hidden layers are included when you publish a SWF file.

To distinguish which layer an object belongs to, display all objects on a layer as colored outlines.

Show or hide a layer or folder

❖ Do one of the following:

- To hide a layer or folder, click in the Eye column to the right of the layer or folder name in the Timeline. To show the layer or folder, click in it again.
- To hide all the layers and folders in the Timeline, click the Eye icon. To show all layers and folders, click it again.
- To show or hide multiple layers or folders, drag through the Eye column.
- To hide all layers and folders other than the current layer or folder, Alt-click (Windows) or Option-click (Macintosh) in the Eye column to the right of a layer or folder name. To show all layers and folders, Alt-click or Option-click it again.

View the contents of a layer as outlines

❖ Do one of the following:

- To display all objects on that layer as outlines, click in the Outline column to the right of the layer's name. To turn off outline display, click in it again.
- To display objects on all layers as outlines, click the outline icon. To turn off outline display on all layers, click it again.
- To display objects on all layers *other* than the current layer as outlines, Alt-click (Windows) or Option-click (Macintosh) in the Outline column to the right of a layer's name. To turn off the outline display for all layers, Alt-click or Option-click in it again.

Change a layer's outline color

1 Do one of the following:

- Double-click the layer's icon (the icon to the left of the layer name) in the Timeline.
- Right-click (Windows) or Control-click (Macintosh) the layer name and select Properties from the context menu.
- Select the layer in the Timeline and select Modify > Timeline > Layer Properties.

2 In the Layer Properties dialog box, click the Outline Color box, select a new color, and click OK.

Change layer height in the Timeline

1 Do one of the following:

- Double-click the layer's icon (the icon to the left of the layer name) in the Timeline.

- Right-click (Windows) or Control-click (Macintosh) the layer name and select Properties from the context menu.
- Select the layer in the Timeline and select Modify > Timeline > Layer Properties.

2 In the Layer Properties dialog box, select an option for Layer Height and click OK.

Change the number of layers displayed in the Timeline

❖ Drag the bar that separates the Timeline from the Stage area.

Edit layers and layer folders

By default, new layers are named by the order in which they are created: Layer 1, Layer 2, and so on. To better reflect their contents, rename layers.

Select a layer or folder

❖ Do one of the following:

- Click the name of a layer or folder in the Timeline.
- Click any frame in the Timeline of the layer to select.
- Select an object on the Stage that is located in the layer to select.
- To select contiguous layers or folders, Shift-click their names in the Timeline.
- To select discontiguous layers or folders, Control-click (Windows) or Command-click (Macintosh) their names in the Timeline.

Rename a layer or folder

❖ Do one of the following:

- Double-click the name of the layer or folder in the Timeline and enter a new name.
- Right-click (Windows) or Control-click (Macintosh) the name of the layer or folder and select Properties from the context menu. Enter the new name in the Name box and click OK.
- Select the layer or folder in the Timeline and select Modify > Timeline > Layer Properties. Enter the new name in the Name box and click OK.

Lock or unlock one or more layers or folders

❖ Do one of the following:

- To lock a layer or folder, click in the Lock column to the right of the name. To unlock the layer or folder, click in the Lock column again.
- To lock all layers and folders, click the padlock icon. To unlock all layers and folders, click it again.
- To lock or unlock multiple layers or folders, drag through the Lock column.
- To lock all *other* layers or folders, Alt-click (Windows) or Option-click (Macintosh) in the Lock column to the right of a layer or folder name. To unlock all layers or folders, Alt-click or Option-click in the Lock column again.

Copy a layer

1 To select the entire layer, click the layer name in the Timeline.

2 To create a layer, click the Insert Layer button.

3 Select Edit > Timeline > Copy Frames.

4 Click the new layer and select Edit > Timeline > Paste Frames.

Copy the contents of a layer folder

1 Collapse the folder (click the triangle to the left of the folder name in the Timeline) and click the folder name to select the entire folder.

2 Select Edit > Timeline > Copy Frames.

3 To create a folder, select Insert > Timeline > Layer Folder.

4 Click the new folder and select Edit > Timeline > Paste Frames.

Delete a layer or folder

1 To select the layer or folder, click its name in the Timeline or any frame in the layer.

2 Do one of the following:

- Click the Delete Layer button in the Timeline.
- Drag the layer or folder to the Delete Layer button.
- Right-click (Windows) or Control-click (Macintosh) the layer or folder name and select Delete Layer from the context menu.

***Note:** When you delete a layer folder, all the enclosed layers and all their contents are also deleted.*

Organize layers and layer folders

To organize your document, rearrange layers and folders in the Timeline.

Layer folders help organize your workflow by letting you place layers in a tree structure. To see the layers a folder contains without affecting which layers are visible on the Stage, expand or collapse the folder. Folders can contain both layers and other folders, allowing you to organize layers in much the same way you organize files on your computer.

The layer controls in the Timeline affect all layers within a folder. For example, locking a layer folder locks all layers within that folder.

- To move a layer or layer folder into a layer folder, drag the layer or layer folder name to the destination layer folder name.
- To change the order of layers or folders, drag one or more layers or folders in the Timeline to the desired position.
- To expand or collapse a folder, click the triangle to the left of the folder name.
- To expand or collapse all folders, Right-click (Windows) or Control-click (Macintosh) and select Expand All Folders or Collapse All Folders.

Use guide layers

For help in aligning objects when drawing, create guide layers and align objects on other layers to the objects you create on the guide layers. Guide layers are not exported and do not appear in a published SWF file. Any layer can be a guide layer. Guide layers are indicated by a guide icon to the left of the layer name.

To control the movement of objects in a motion tweened animation, create a motion guide layer.

Note: Dragging a normal layer onto a guide layer converts the guide layer to a motion guide layer. To prevent accidentally converting a guide layer, place all guide layers at the bottom of the layer order.

❖ Select the layer and Right-click (Windows) or Control-click (Macintosh) and select Guide from the context menu. To change the layer back to a normal layer, select Guide again.

See also

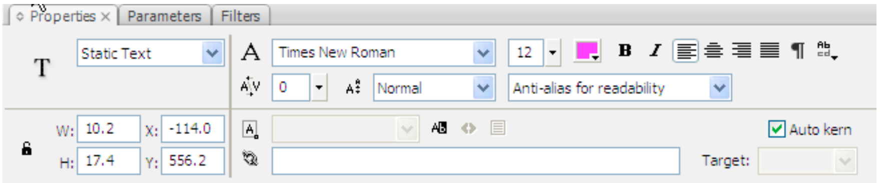
"Tween motion along a path" on page 244

Using Flash authoring panels

About the Property inspector

The Property inspector provides easy access to the most commonly used attributes of the current selection, either on the Stage or in the Timeline. You can make changes to the object or document attributes in the Property inspector without accessing the menus or panels that also control these attributes.

Depending on what is currently selected, the Property inspector displays information and settings for the current document, text, symbol, shape, bitmap, video, group, frame, or tool. When two or more different types of objects are selected, the Property inspector displays the total number of objects selected.



Static Text

Times New Roman

12

0

Normal

Anti-alias for readability

W: 10.2 X: -114.0

H: 17.4 Y: 556.2

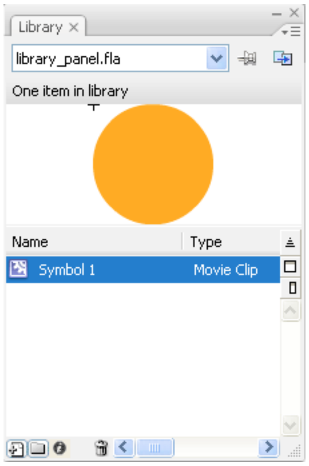
☑ Auto kern

The Property inspector showing the properties for the Text tool

To display the Property inspector, Select Window > Properties > Properties, or press Control+F3 (Windows) or Command+F3 (Macintosh).

About the Library panel

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.



The Library panel showing a movie clip symbol

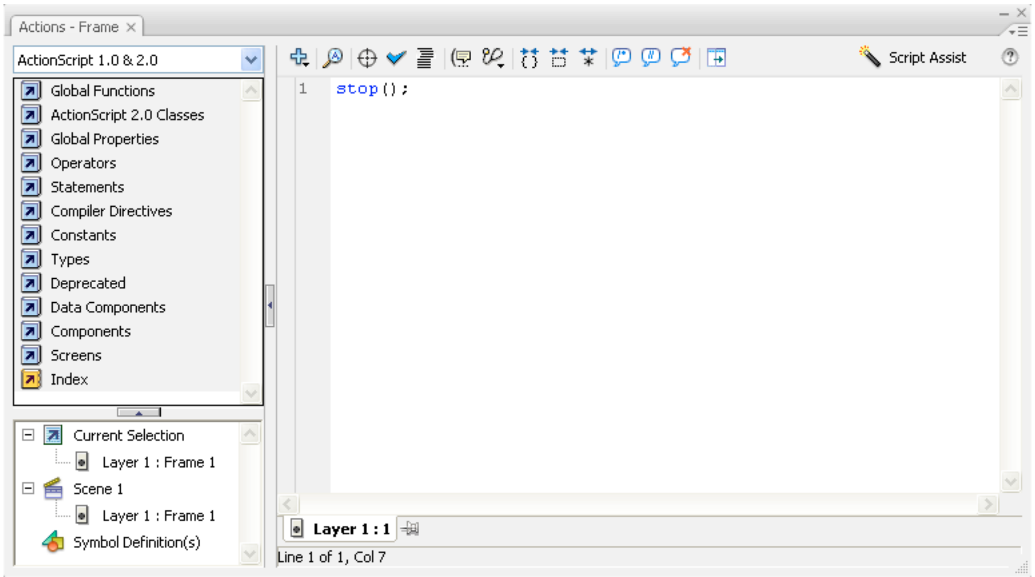
To display the Library panel, select Window > Library, or press Control+L (Windows) or Command+L (Macintosh).

See also

“Managing media assets with the Flash document library” on page 64

About the Actions panel

The Actions panel lets you create and edit ActionScript code 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.



```
stop();
```

The Actions panel showing a stop() action in a frame

To display the Actions panel, select Window > Actions or press F9.

See also

"Actions panel overview" on page 382

"Script window overview" on page 383

Use the Movie Explorer

The Movie Explorer lets you view and organize the contents of a document and select elements in the document for modification. It contains a display list of currently used elements, arranged in a navigable hierarchical tree.

Use the Movie Explorer to perform the following actions:

- Filter which categories of items in the document appear in the Movie Explorer.
- Display the selected categories as scenes, symbol definitions, or both.
- Expand and collapse the navigation tree.
- Search for an element in a document by name.
- Familiarize yourself with the structure of a Flash document that another developer created.
- Find all the instances of a particular symbol or action.
- Print the navigable display list that appears in the Movie Explorer.

The Movie Explorer has a Panel menu and a context menu with options for performing operations on selected items or modifying the Movie Explorer display. A check mark with a triangle below it in the Movie Explorer panel indicates the Panel menu.

***Note:** The Movie Explorer has slightly different functionality when you are working with screens.*

See also

"Working with screens" on page 366

View the Movie Explorer

❖ Select Window > Movie Explorer.

Filter the categories of items that appear in the Movie Explorer

- To show text, symbols, ActionScript, imported files, or frames and layers, click one or more of the filtering buttons to the right of the Show option. To customize which items to show, click the Customize button. Select options in the Show area of the Movie Explorer Settings dialog box to view those elements.
- To show items in scenes, select Show Movie Elements from the Movie Explorer Panel menu.
- To show information about symbols, select Show Symbol Definitions from the Movie Explorer Panel menu.

***Note:** The Movie Elements option and the Symbol Definitions option can be active at the same time.*

Search for an item using the Find box

❖ In the Find box, enter the item name, font name, ActionScript string, or frame number. The Find feature searches all items that appear in the Movie Explorer.

Select an item in the Movie Explorer

❖ Click the item in the navigation tree. Shift-click to select more than one item.

The full path for the selected item appears at the bottom of the Movie Explorer. Selecting a scene in the Movie Explorer shows the first frame of that scene on the Stage. Selecting an element in the Movie Explorer selects that element on the Stage if the layer containing the element is not locked.

Use the Movie Explorer Panel menu or context menu commands

1 Do one of the following:

- To view the Panel menu, click the Panel menu control in the Movie Explorer panel.
- To view the context menu, right-click (Windows) or Control-click (Macintosh) an item in the Movie Explorer navigation tree.

2 Select an option from the menu:

Go To Location Jumps to the selected layer, scene, or frame in the document.

Go To Symbol Definition Jumps to the symbol definition for a symbol that is selected in the Movie Elements area of the Movie Explorer. The symbol definition lists all the files associated with the symbol. (The Show Symbol Definitions option must be selected. See its definition in this list.)

Select Symbol Instances Jumps to the scene containing instances of a symbol that is selected in the Symbol Definitions area of the Movie Explorer. (The Show Movie Elements option must be selected.)

Find In Library Highlights the selected symbol in the document's library. (Flash opens the Library panel if it is not already visible.)

Rename Lets you enter a new name for a selected element.

Edit In Place Lets you edit a selected symbol on the Stage.

Edit In New Window Lets you edit a selected symbol in a new window.

Show Movie Elements Shows the elements in your document organized into scenes.

Show Symbol Definitions Shows all the elements associated with a symbol.

Copy All Text To Clipboard Copies selected text to the clipboard. For spell checking or other editing, paste the text into an external text editor.

Cut, Copy, Paste, And Clear Performs these common functions on a selected element. Modifying an item in the display list modifies the corresponding item in the document.

Expand Branch Expands the navigation tree at the selected element.

Collapse Branch Collapses the navigation tree at the selected element.

Collapse Others Collapses the branches in the navigation tree that do not contain the selected element.

Print Prints the hierarchical display list that appears in the Movie Explorer.

About the Web Services panel

You can view a list of web services, refresh web services, and add or remove web services in the Web Services panel (Window > Other Panels > Web Services). When you add a web service to the Web Services panel, the web service is then available to any application you create.

You can use the Web Services panel to refresh all your web services at once by clicking the Refresh Web Services button. If you are not using the Stage but instead are writing ActionScript code for the connectivity layer of your application, you can use the Web Services panel to manage your web services.

For detailed information about using the web services panel, see www.adobe.com/go/learn_fl_web_services.

Accessibility in the Flash workspace

About accessibility support

Accessibility support in the authoring environment provides keyboard shortcuts for navigating and using interface controls, including panels, the Property inspector, dialog boxes, the Stage, and objects on the Stage, so that you can work with these interface elements without using the mouse.

***Note:** Certain keyboard controls and authoring environment accessibility features are available only in Windows.*

To customize the keyboard shortcuts for accessibility in the authoring environment, use the Workspace Accessibility Commands section of the Keyboard Shortcuts dialog box.

See also

"Customize keyboard shortcuts" on page 31

About Flash authoring accessibility on the Macintosh

Accessibility for the authoring environment on the Macintosh has the following limitations:

- The Panel Focus keyboard shortcut (Command+Option+Tab) is not supported for the Property inspector.
- The Panel Control Focus keyboard shortcut (Tab) is supported only for the Timeline, not for other panels or the Property inspector.

Select panels or the Property inspector with keyboard shortcuts

To select a panel or the Property inspector (also referred to as applying *focus* to the panel or Property inspector), use the keyboard shortcut Control+F6 (Windows) or Command+F6 (Macintosh).

Apply focus to a panel or the Property inspector only when the panel or Property inspector is visible in the application window. The panel can be expanded or collapsed.

When you use the keyboard shortcut to select panels, focus is applied to panels using the following criteria:

- Docked panels are given focus first.
- If the Timeline is showing and docked, the Timeline is given focus the first time you press Control+F6 (Windows) or Command+F6 (Macintosh).
- If the Timeline is not showing and docked, or if you press the keyboard shortcut again, focus moves to the rightmost and highest docked panel. Pressing the keyboard shortcut repeatedly then moves the focus through the other docked panels, from right to left and from top to bottom of the workspace.
- If you move the focus through all the docked panels, or if no docked panels are showing, focus moves to the rightmost and highest floating panel. Pressing the keyboard shortcut repeatedly then moves the focus through the other floating panels, from right to left and from top to bottom of the workspace.

Use keyboard shortcuts to select or deselect, expand, or collapse panels or the Property inspector

- To move the focus through the panels currently displayed in the workspace, press Control+F6 (Windows) or Command+F6 (Macintosh). A dotted line appears around the title of the currently focused panel.
- To move the focus to the previously selected panel, press Control+Shift+F6 (Windows) or Command+Shift+F6 (Macintosh).

- To deselect a panel, press Escape, or move, dock, or undock the panel.
- To move the focus to the panel above or below the current panel in a panel group, press Up Arrow or Down Arrow.
- To hide all panels and the Property inspector, press F4. To display all panels and the Property inspector, press F4 again.

Use keyboard shortcuts to expand or collapse panels or the Property inspector

1 Press Control+F6 (Windows) or Command+F6 (Macintosh) until the panel to expand or collapse has focus. A dotted line appears around the title of the currently focused panel.

2 To expand or collapse the currently selected panel, press the Spacebar.

Select controls in a panel or the Property inspector using keyboard shortcuts

To move the focus through the panel controls when a panel or the Property inspector has the current focus, use the Tab key. To activate the control that has the current focus, use the Spacebar (that is, pressing Spacebar is equivalent to clicking a control in the panel).

When you use the keyboard shortcut for panel controls, focus is applied to a control and the control is activated using the following criteria:

- To select a control in the panel with the Tab key, the panel with the current focus must be expanded. If the panel is collapsed, pressing Tab has no effect.
- When the panel with the current focus is expanded, pressing Tab the first time moves the focus to the panel's Panel menu.
- To move the focus between the Panel menu and the panel title bar, use Right Arrow and Left Arrow.
- If the focus is on the Panel menu, press Tab again to move the focus through the other controls in the panel. Pressing Tab again does not return the focus to the Panel menu.
- To display the Panel menu items when the Panel menu has the focus, press Enter (Windows only).
- To move the focus between the Panel menus of the panels in the group in panels that are grouped, use Up Arrow and Down Arrow.
- You can move the focus to a panel control only if the control is active. If a control is dimmed (inactive), you cannot apply focus to the control.

Move the focus from a panel title bar to a panel options menu

❖ Do one of the following:

- Press Tab.
- Press Right Arrow. To return the focus to the panel title bar, press Left Arrow or Shift+Tab.
- To move the focus to the Panel menu of the panel immediately above the panel with the current focus if the panel is in a group, press Up Arrow. To move the focus to the Panel menu of the panel immediately below the panel with the current focus, press Down Arrow.

Move the focus through the items in the Panel menu of a panel

1 To display the Panel menu items with the focus currently applied to the Panel menu, press the Spacebar.

2 To move through the items in the Panel menu, press Down Arrow.

3 To activate the currently selected Panel menu item, press Enter (Windows) or Return (Macintosh).

Move the focus through the controls in a panel

1 Press Tab when the focus is currently applied to the Panel menu. To move the focus through the controls in the panel, press Tab repeatedly.

2 To activate the currently selected panel control, press Enter (Windows only).

Navigate dialog box controls using keyboard shortcuts (Windows only)

- To move through the controls in the dialog box, press Tab.
- To move through the controls within one section of a dialog box, press Up Arrow and Down Arrow.
- To activate the button (equivalent to clicking the button), when the focus is applied to a dialog box control button, press Enter.
- To apply the current settings and close the dialog box (equivalent to clicking OK), when the focus is not applied to any dialog box control button, press Enter.
- To close the dialog box without applying the changes (equivalent to clicking Cancel), press Escape.
- To view the Help content for the dialog box (equivalent to clicking Help), when the focus is applied to the Help button, press Enter or Spacebar.

Select the Stage or objects on the Stage using keyboard shortcuts

Selecting the Stage with a keyboard shortcut is equivalent to clicking on the Stage. Any other element currently selected becomes deselected when the Stage is selected.

After the Stage is selected, use the Tab key to navigate through all objects on all layers, one at a time. You can select instances (including graphic symbols, buttons, movie clips, bitmaps, videos, or sounds), groups, or boxes. You cannot select shapes (such as rectangles) unless those shapes are instances of symbols. You cannot select more than one object at a time using keyboard shortcuts.

To select Objects on the Stage, use the following criteria:

- To select the previous object when an object is currently selected, press Shift+Tab.
- To select the first object that was created on the active frame in the active layer, press Tab. When the last object on the top layer is selected, press Tab to move to the next layer beneath it and select the first object there, and so on.
- When the last object on the last layer is selected, press Tab to move to the next frame and select the first object on the top layer there.
- Objects on layers that are hidden or locked cannot be selected with the Tab key.
- To select the Stage, press Control+Alt+Home (Windows) or Command+Option+Home (Macintosh).
- To select an object on the Stage, with the Stage selected, press Tab.

Note: *If you are currently typing text in a box, you cannot select an object using the keyboard focus. You must first change the focus to the Stage and then select an object.*

Navigate tree structures using keyboard shortcuts

To navigate *tree structures,* the hierarchical displays of file structures in certain Flash panels, use keyboard shortcuts.

- To expand a collapsed folder, select the folder and press Right Arrow.
- To collapse an expanded folder, select the folder and press Left Arrow.
- To move to the parent folder of an expanded folder, press Left Arrow.

- To move to the child folder of an expanded folder, press Right Arrow.

Work with library items using keyboard shortcuts

1 To copy or paste a selected library item, press Control+X (Windows) or Command+X (Macintosh) to cut the item, or press Control+C (Windows) or Command+C (Macintosh) to copy the item.

2 To paste a cut or copied item, click the Stage or in another library to set the insertion point, and press Control+V (Windows) or Command+V (Macintosh) to paste in the center of the Stage; or press Control+Shift+C (Windows) or Command+Shift+C (Macintosh) to paste in place (in the same location as the original).

To cut, copy, and paste items, use the following criteria:

- Cut or copy one item or multiple items.
- Cut or copy an item from the Library panel and paste it onto the Stage or into another library, or paste a folder into another library.
- You cannot paste a shape from the Stage into the library.
- You cannot paste a library item into a common library, because common libraries cannot be modified. However, you can create a common library.
- When you paste a library item onto the Stage, the item is centered.
- If you paste a folder, each item in the folder is included.
- To paste a library item into a folder in the destination library, click the folder before pasting.
- You can paste a library item into a different location in the same library where it originated.
- If you attempt to paste a library item into a location containing another item by the same name, select whether to replace the existing item.

See also

“Work with common libraries” on page 68

Undo, redo, and history

Undo, Redo, and Repeat commands

To undo or redo actions on individual objects, or all objects within the current document, specify either object-level or document-level Undo and Redo commands (Edit > Undo or Edit Redo). The default behavior is document-level Undo and Redo.

You cannot undo some actions when using object-level Undo. Among these are entering and exiting Edit mode; selecting, editing, and moving library items; and creating, deleting, and moving scenes.

To remove deleted items from a document after using the Undo command, use the Save And Compact command.

To reapply a step to the same object or to a different object, use the Repeat command. For example, if you move a shape named shape_A, select Edit > Repeat to move the shape again, or select another shape, shape_B, and select Edit > Repeat to move the second shape by the same amount.

By default, Flash supports 100 levels of undo for the Undo menu command. Select the number of undo and redo levels, from 2 to 9999, in Flash Preferences.

See also

"Set preferences in Flash" on page 27

"Automating tasks with the Commands menu" on page 49

Permanently remove items deleted with Undo

By default, when you undo a step using Edit > Undo or the History panel, the file size of the document does not change, even if you delete an item in the document. For example, if you import a video file into a document, and undo the import, the file size of the document still includes the size of the video file. Any items that you delete from a document when performing an Undo command are preserved to restore the items with a Redo command. To permanently remove the deleted items from the document, and reduce the document file size, select File > Save And Compact.

Using the History panel

The History panel (Window > Other Panels > History) shows a list of the steps you've performed in the active document since you created or opened that document, up to a specified maximum number of steps. (The History panel doesn't show steps you've performed in other documents.) The slider in the History panel initially points to the last step that you performed.

To undo or redo individual steps or multiple steps at once, use the History panel. Apply steps from the History panel to the same object or to a different object in the document. However, you cannot rearrange the order of steps in the History panel. The History panel is a record of steps in the order in which they are performed.

***Note:** If you undo a step or a series of steps and then do something new in the document, you can no longer redo the steps in the History panel; they disappear from the panel.*

To remove deleted items from a document after you undo a step in the History panel, use the Save And Compact command.

By default, Flash supports 100 levels of undo for the History panel. Select the number of undo and redo levels, from 2 to 9999, in Flash Preferences.

To erase the history list for the current document, clear the History panel. After clearing the history list, you cannot undo the steps that are cleared. Clearing the history list does not undo steps; it removes the record of those steps from the current document's memory.

Closing a document clears its history. To use steps from a document after that document is closed, copy the steps with the Copy Steps command or save the steps as a command.

See also

"Set preferences in Flash" on page 27

"Automating tasks with the Commands menu" on page 49

Undo steps with the History panel

When you undo a step, the step is dimmed in the History panel.

- To undo the last step performed, drag the History panel slider up one step in the list.
- To undo multiple steps at once, drag the slider to point to any step, or click to the left of a step along the path of the slider. The slider scrolls automatically to that step, undoing all subsequent steps as it scrolls.

***Note:** Scrolling to a step (and selecting the subsequent steps) is different from selecting an individual step. To scroll to a step, click to the left of the step.*

Replay steps with the History panel

When you replay steps with the History panel, the steps that play are the steps that are selected (highlighted) in the History panel, not necessarily the step currently indicated by the slider.

Apply steps in the History panel to any selected object in the document.

Replay one step

❖ In the History panel, select a step and click the Replay button.

Replay a series of adjacent steps

1 Select steps in the History panel by doing one of the following:

- Drag from one step to another. (Don't drag the slider; drag from the text label of one step to the text label of another step.)
- Select the first step, then Shift-click the last step; or select the last step and Shift-click the first step.

2 Click Replay. The steps replay in order, and a new step, labeled Replay Steps, appears in the History panel.

Replay nonadjacent steps

1 Select a step in the History panel, and Control-click (Windows) or Command-click (Macintosh) other steps. To deselect a selected step, Control-click or Command-click.

2 Click Replay.

Copy and paste steps between documents

Each open document has its own history of steps. To copy steps from one document and paste them into another, use the Copy Steps command in the History panel options menu. If you copy steps into a text editor, the steps are pasted as JavaScript code.

1 In the document containing the steps to reuse, select the steps in the History panel.

2 In the History panel options menu, select Copy Steps.

3 Open the document to paste the steps into.

4 Select an object to apply the steps to.

5 Select Edit > Paste to paste the steps. The steps play back as they're pasted into the document's History panel. The History panel shows them as only one step, called Paste Steps.

Automating tasks with the Commands menu

Create and manage commands

To repeat the same task, create a command in the Commands menu from steps in the History panel and reuse the command. Steps replay exactly as they were originally performed. You can't modify the steps as you replay them.

To use steps the next time you start Flash, create and save a command. Saved commands are retained permanently, unless you delete them. Steps that you copy using the History panel Copy Steps command are discarded when you copy something else.

Create a command from selected steps in the History panel. Rename or delete commands in the Manage Saved Commands dialog box.

See also

"Copy and paste steps between documents" on page 49

Create a command

1 Select a step or set of steps in the History panel.

2 Select Save As Command from the History panel options menu.

3 Enter a name for the command and click OK. The command appears in the Commands menu.

Note: *The command is saved as a JavaScript file (with the extension .jsfl) in your Commands folder. This folder is in the following locations: Windows 2000 or Windows XP: boot drive\Documents and Settings\<user>\Local Settings\Application Data\Adobe\Flash CS3\<language>\Configuration\Commands; Mac OS X: Macintosh HD/Users/<username>/Library/Application Support/Adobe/Flash CS3/<language>/Configuration/Commands.*

Edit the names of commands in the Commands menu

1 Select Commands > Manage Saved Commands.

2 Select a command to rename. enter a new name for it, and click Close.

Delete a name from the Commands menu

1 Select Commands > Manage Saved Commands, and select a command.

2 Click Delete, and click Close.

Run commands

- To use a saved command, select the command from the Commands menu.
- To run a JavaScript or Flash JavaScript command, select Commands > Run Command, navigate to the script to run, and click Open.

Get more commands

Use the Get More Commands option in the Commands menu to link to the Flash Exchange website at www.adobe.com/go/flash_exchange and download commands that other Flash users have posted. For more information on the commands posted there, see the Flash Exchange website.

1 Make sure you are connected to the Internet.

2 Select Commands > Get More Commands.

Steps that can't be used in commands

Some tasks can't be saved as commands or repeated using the Edit > Repeat menu item. These commands can be undone and redone, but they cannot be repeated.

Examples of actions that can't be saved as commands or repeated include selecting a frame or modifying a document size. If you attempt to save an unrepeatable action as a command, the command is not saved.

Chapter 3: Creating and managing documents

When you create and save Adobe® Flash® CS3 Professional documents within the Flash authoring environment, the documents are in FLA file format. To display a document in Adobe® Flash® Player, you must publish or export the document as a SWF file.

You can add media assets to a Flash document and manage the assets in the library, and you can use the Movie Explorer to view and organize all the elements in a Flash document. The Undo and Redo commands, the History panel, and the Commands menu let you automate tasks in a document.

Working with Flash documents

About Flash files

In Flash you can work with a variety of file types, each of which has a separate purpose:

- FLA files, the primary files you work with in Flash, contain the basic media, timeline, and script information for a Flash document. *Media objects* are the graphic, text, sound, and video objects that comprise the content of your Flash document. The *Timeline* is where you tell Flash when specific media objects should appear on the Stage. You can add *ActionScript*™ code to Flash documents to more finely control their behavior and to make them respond to user interactions.
- SWF files, the compiled versions of FLA files, are the files you display in a web page. When you publish your FLA file, Flash creates a SWF file.
- AS files are ActionScript files—you can use these to keep some or all of your ActionScript code outside of your FLA files, which is helpful for code organization and for projects that have multiple people working on different parts of the Flash content.
- SWC files contain the reusable Flash components. Each SWC file contains a compiled movie clip, ActionScript code, and any other assets that the component requires.
- ASC files are files used to store ActionScript that will be executed on a computer running Flash Media Server. These files provide the ability to implement server-side logic that works in conjunction with ActionScript in a SWF file.
- JSFL files are JavaScript files that you can use to add new functionality to the Flash authoring tool.
- FLP files are Flash project files. You can use Flash projects to manage multiple document files in a single project. Flash projects allow you to group multiple, related files together to create complex applications.

For video tutorials about working with Flash files, see the following:

- www.adobe.com/go/vid0117
- www.adobe.com/go/vid0118

See also

“About the Timeline” on page 33

Create or open a document and set its properties

You can create a new document or open a previously saved document in Flash, and you can open a new window as you work. You can set properties for new or existing documents.

For a text tutorial about creating your first Flash file, see Create your First File on the Flash Tutorials page at www.adobe.com/go/learn_fl_tutorials.

For video tutorials, see:

- Working with Flash files: www.adobe.com/go/vid0117

See also

“Set preferences in Flash” on page 27

“Publishing Flash content” on page 418

Create a new document

1 Select File > New.

2 On the General tab, select Flash Document.

Create a new document of the same type as the last document created (Windows only)

❖ Click the New File button in the main toolbar.

Create a new document from a template

1 Select File > New.

2 Click the Templates tab.

3 Select a category from the Category list, select a document from the Category Items list, and click OK. You can select standard templates that come with Flash or a template you have already saved.

Open an existing document

1 Select File > Open.

2 In the Open dialog box, navigate to the file or enter the path to the file in the Go To box.

3 Click Open.

Open a new window in the current document

❖ Select Window > Duplicate Window.

Set properties for a new or existing document

1 With the document open, select Modify > Document.

The Document Properties dialog box appears.

2 To embed metadata within your SWF files, enter a descriptive title in the Title box, and enter a description in the Description box.

Embedding metadata improves the ability of web-based search engines to return meaningful search results for Flash content. Descriptions can author and copyright information, and short descriptions about the content and its purpose. The search metadata is based on the RDF (Resource Description Framework) and XMP (Extensible Metadata Platform) specifications and is stored in Flash in a W3C-compliant format.

The title and description you enter is a human readable title and a human readable description. These fields are not intended for keywords to provide greater search results. Instead, these fields are made available to search engines that index SWF files, and display the contents of the title and description field in their search results. Search metadata can be exported to any version of Flash. While it was introduced in Flash 8, Flash Player ignores tags it does not understand, thus Flash 8 exports it to all versions.

***Note:** Flash lets you make the settings you specify in the Document Properties dialog box the default settings for any Flash document that you create. The exception to this is the Title and Description, which you need to specify for each Flash document that you create.*

3 For Frame Rate, enter the number of animation frames to appear every second.

For most computer-displayed animations, especially those playing from a website, 8 frames per second (fps) to 12 fps (the default) is sufficient.

4 For Dimensions, set the Stage size:

- To specify the Stage size in pixels, enter values in the Width and Height boxes. The minimum size is 1 x 1 pixels; the maximum is 2880 x 2880 pixels.
- To set the Stage size so that there is equal space around the content on all sides, click the Contents button to the right of Match. To minimize document size, align all elements to the upper-left corner of the Stage, and then click Contents.
- To set the Stage size to the maximum available print area, click Printer. This area is determined by the paper size minus the current margin selected in the Margins area of the Page Setup dialog box (Windows) or the Print Margins dialog box (Macintosh).
- To set the Stage size to the default size, 550 x 400 pixels, click Default.

5 To set the background color of your document, click the triangle in the Background Color control and select a color from the palette.

6 To specify the unit of measure for rulers that you can display along the top and side of the application window, select an option from the pop-up menu in the upper right. (This setting also determines the units used in the Info panel.)

7 Do one of the following:

- To make the new settings the default properties for the current document only, click OK.
- To make the new settings the default properties for all new documents, click Make Default.

Change document properties using the Property inspector

1 Deselect all assets, then select the Selection tool.

2 In the Property inspector (Window > Properties > Properties), click the Size control to display the Document Properties dialog box.

3 To select a background color, click the triangle in the Background color control and select a color from the palette.

4 For Frame Rate, enter the number of animation frames to appear every second.

5 For Publish, click the Settings button to display the Publish Settings dialog box with the Flash tab selected. For more information, see “Publishing Flash documents” on page 418.

6 If you are developing content for mobile devices such as cell phones, click the Settings button to display the Device Settings dialog box, which lets you choose devices to test mobile content and provides information on ActionScript support for each device you select.

***Note:** The Device Settings button can be used only if your publish settings are set to a supported version of Flash Lite.*

View a document when multiple documents are open

When you open multiple documents, tabs at the top of the Document window identify the open documents and let you easily navigate among them. Tabs appear only when documents are maximized in the Document window.

❖ Click the tab of the document you want to view.

By default, tabs appear in the order in which the documents were created. You can drag the document tabs to change their order.

Save Flash documents

You can save a Flash FLA document using its current name and location or using a different name or location.

When a document contains unsaved changes, an asterisk (*) appears after the document name in the document title bar, the application title bar, and the document tab. When you save the document, the asterisk is removed.

Save a Flash document

1 Do one of the following:

- To overwrite the current version on the disk, select File > Save.
- To save the document in a different location and/or with a different name, or to compress the document, select File > Save As.

2 If you selected Save As, or if the document has never been saved before, enter the filename and location.

3 Click Save.

Revert to the last saved version of a document

❖ Select File > Revert.

Save a document as a template

1 Select File > Save As Template.

2 In the Save As Template dialog box, enter a name for the template in the Name box.

3 Select a category from the Category pop-up menu, or enter a name to create a new category.

4 Enter a description of the template in the Description box (up to 255 characters), and click OK.

The description appears when the template is selected in the New Document dialog box.

Save a document as a Flash 8 document

1 Select File > Save As.

2 Enter the filename and location.

3 Select Flash 8 Document from the Format pop-up menu, and click Save.

***Important:** If an alert message indicates that content will be deleted if you save in Flash 8 format, click Save As Flash 8 to continue. This might happen if your document contains features, such as graphic effects or behaviors, that are available only in Flash 9. Flash does not preserve these features when you save the document in Flash 8 format.*

Save documents when quitting Flash

1 Select File > Exit (Windows) or Flash > Quit Flash (Macintosh).

2 If you have documents open with unsaved changes, Flash prompts you to save or discard the changes for each document.

- Click Yes to save the changes and close the document.
- Click No to close the document without saving the changes.

Working with other Adobe applications

Flash is designed to work with other Adobe applications to enable a broad range of creative workflows. You can import Illustrator and Photoshop files directly into Flash. You can also create video from Flash and edit it in Premier Pro or After Effects, or import video from either of those applications into Flash. When publishing your Flash content, you can use Dreamweaver to embed the content in your web pages, and launch Flash directly from within Dreamweaver to edit the content.

See also

"Working with Illustrator and Flash" on page 134

"Working with Photoshop and Flash" on page 147

"Working with Premier Pro and After Effects" on page 316

"Edit a SWF file from Dreamweaver in Flash" on page 419

Creating and previewing mobile content with Adobe Device Central

Access Adobe components from Adobe Device Central

1 Start Device Central.

2 Select File > New Document In > Flash, Illustrator, or Photoshop.

In Device Central, the New Document panel appears with the correct options to create a new mobile document in the selected application.

3 Make any necessary changes, such a selecting a new Player Version, ActionScript Version, or Content type.

4 Do one of the following:

- Select the Custom Size for All Selected Devices option and add a width and height (in pixels).
- Select a device or multiple devices from the Device Sets list or Available Devices list.

5 If you selected multiple devices, Device Central selects a size for you. If you want to select a different size, click on a different device or set of devices.

6 Click Create.

The selected application opens with a new mobile document ready to edit.

Create mobile content with Adobe Device Central and Flash CS3

1 Start Flash.

2 On the main Flash screen, select Create New > Flash File (Mobile).

Flash opens Device Central and displays the New Document tab.

3 In Device Central, select a Player version and ActionScript version.

The Available Devices list on the left is updated. Devices that do not support the selected Player version and ActionScript version are dimmed.

4 Select a content type.

The Available Devices list on the left is updated and shows the devices that support the content type (as well as the Player version and ActionScript version) selected.

5 In the Available Devices list, select a single target device or multiple devices (or select a set or individual device in the Device Sets list).

Device Central lists proposed document sizes based on the device or devices you selected (if the devices have different display sizes). Depending on the design or content you are developing, you can create a separate mobile document for each display size or try to find one size appropriate for all devices. When choosing the second approach, you may want to use the smallest or largest suggested document size as a common denominator. You can even specify a custom size at the bottom of the tab.

6 Click Create.

Flash starts up and creates a document with preset publish settings from Device Central, including the correct size for the device (or group of devices) specified.

7 Add content to the new Flash document.

8 To test the document, select Control > Test Movie.

The new document is displayed in the Device Central Emulator tab. If one or more devices were selected in the Available Devices list in step 5, a new device set is created (named according to the FLA file) and listed in the Device Sets panel. The device shown in the Emulator tab is listed in the Device Sets panel with a special icon ◐. To test the new Flash document on another device, double-click the name of a different device in the Device Sets or Available Devices lists.

For tutorials about creating content using Flash and Device Central, see http://www.adobe.com/go/vid0186 and http://www.adobe.com/go/vid0206.

Create mobile content with Adobe Device Central and Photoshop

1 Start Photoshop.

2 Select File > New.

3 Click Device Central to close the dialog box in Photoshop and open Device Central.

4 Select a content type.

The available Devices list on the left is updated and shows the devices that support the content type selected.

5 In the Available Devices list, select a single target device or multiple devices (or select a set or individual device in the Device Sets list).

Device Central lists proposed document sizes based on the device or devices you selected (if the devices have different display sizes). Depending on the design or content you are developing, you can create a separate mobile document for each display size or try to find one size appropriate for all devices. When choosing the second approach, you may want to use the smallest or largest suggested document size as a common denominator. You can even specify a custom size at the bottom of the tab.

6 Click Create.

A blank PSD file with the specified size opens in Photoshop. The new file has the following parameters set by default:

- Color Mode: RGB/8bit
- Resolution: 72 ppi
- Color Profile: SRGB IEC61966-2.1

7 Fill the blank PSD file with content in Photoshop.

8 When you finish, select File > Save For Web & Devices.

9 In the Save For Web & Devices dialog box, select the desired format and change other export settings as desired.

10 Click Device Central.

A temporary file with the export settings specified is displayed in the Device Central Emulator tab. To continue testing, double-click the name of a different device in the Device Sets or Available Devices lists.

11 If, after previewing the file in Device Central, you need to make changes to the file, go back to Photoshop.

12 In the Photoshop Save For Web & Devices dialog box, make adjustments, such as selecting a different format or quality for export.

13 To test the file again with the new export settings, click the Device Central button.

14 When you are satisfied with the results, click Save in the Photoshop Save For Web & Devices dialog box.

Note: *To simply open Device Central from Photoshop (instead of creating and testing a file), select File > Device Central.*

For a tutorial about creating content using Photoshop and Device Central, see http://www.adobe.com/go/vid0185.

Create mobile content with Adobe Device Central and Illustrator

1 Start Illustrator.

2 Select File > New.

3 In New Document Profile, select Mobile and Devices.

4 Click Device Central to close the dialog box in Illustrator and open Device Central.

5 Select a content type.

The available Devices list on the left is updated and shows the devices that support the content type selected.

6 In Device Central, select a device, several devices, or a device set.

Based on the device(s) selected and content type, Device Central suggests one or multiple artboard sizes to be created. To create one document at a time, select a suggested document size (or select the Custom Size for all selected Devices option and enter custom values for Width and Height).

7 Click Create.

A blank AI file of the specified size opens in Illustrator. The new file has the following parameters set by default:

- Color Mode: RGB

- Raster Resolution: 72 ppi

8 Fill the blank AI file with content in Illustrator.

9 When you finish, select File > Save For Web & Devices.

10 In the Save for Web & Devices dialog, select the desired format and change other export settings as desired.

11 Click Device Central.

A temporary file with the export settings specified is displayed in the Device Central Emulator tab. To continue testing, double-click the name of a different device in the Device Sets or Available Devices lists.

12 If, after previewing the file in Device Central, you need to make changes to the file, go back to Illustrator.

13 On the Illustrator Save for Web & Devices dialog, make adjustments such as selecting a different format or quality for export.

14 To test the file again with the new export settings, click the Device Central button.

15 When you are satisfied with the results, click Save in the Illustrator Save for Web & Devices dialog.

***Note:** To simply open Device Central from Illustrator (instead of creating and testing a file), select File > Device Central.*

For a tutorial about creating content with Illustrator and Device Central, see http://www.adobe.com/go/vid0207.

Preview a movie on a virtual mobile device using Adobe Premiere Pro

Using Adobe Device Central, you can preview movies formatted for mobile devices in emulations of those devices. This option is available for most of the H.264 formats listed in the Adobe Media Encoder.

1 On Windows computers, make sure QuickTime is installed.

2 Start Adobe Premiere Pro.

3 Open the file to preview.

4 Select the file in the project area or Timeline.

5 Choose File > Export > Adobe Media Encoder.

6 In the Export Settings area of the Export Settings Window, select H.264 from the Format drop-down menu.

7 Select a mobile preset (e.g., 3GPP).

Open in Device Central should be checked by default.

8 Click OK.

9 Name and save the file.

The file is rendered.

10 A temporary file is displayed in the Device Central Emulator tab. To continue testing, double-click the name of a different device in the Device Sets or Available Devices lists.

Preview a movie on a virtual mobile device using After Effects

Using Adobe Device Central, you can preview movies formatted for mobile devices in emulations of those devices. This option is available for most of the H.264 formats listed in the Adobe Media Encoder.

1 Start After Effects.

2 In the Project panel, select the composition to preview.

3 Choose Composition > Add to Render Queue.

4 In the Render Queue panel, click the underlined text to the right of Output Module, or select Custom from the Output Module menu.

5 In the Output Modules Settings dialog box, choose H.264 from the Format menu.

6 In the Export Settings section of the H.264 dialog box, select Open in Device Central.

7 Modify other settings as desired and click OK.

8 Click OK to close the Output Module Settings dialog box.

9 In the Render Queue panel, click Render.

Rendering may take a few minutes, depending on the size of the file. When rendering is complete, a temporary file is displayed in the Adobe Device Central Emulator tab. To continue testing, double-click the name of a different device in the Device Sets or Available Devices lists.

Preview mobile content with Adobe Device Central and Dreamweaver

To preview pages created in Dreamweaver on various mobile devices, use Device Central with its built-in Opera Small-Screen Rendering feature. Different devices have different browsers installed, but the preview can give a good impression of how content will look and behave on a selected device.

1 Start Dreamweaver.

2 Open a file.

3 Do one of the following:

- Select File > Preview in Browser > Device Central.
- On the document window toolbar, click and hold the Preview/Debug In browser button 🌐 and select Preview In Device Central.

The file is displayed in the Device Central Emulator tab. To continue testing, double-click the name of a different device in the Device Sets or Available Devices lists.

Access Adobe Device Central from Adobe Bridge

To access Device Central from Adobe Bridge, select an individual file. The supported formats are: SWF, JPG, JPEG, PNG, GIF, WBM, MOV, 3GP, M4V, MP4, MPG, MPEG, AVI, HTM, HTML, XHTML, CHTML, URL, and WEBLOC.

1 Start Adobe Bridge.

2 Do one of the following:

- Select a file and click File > Test in Device Central.
- Right-click a file and select Test in Device Central.

The file is displayed in the Device Central Emulator tab. To continue testing, double-click the name of a different device in the Device Sets or Available Devices lists.

***Note:** To browse device profiles or to create mobile documents, select Tools > Device Central. Device Central opens with the Devices Profiles tab shown.*

For a tutorial about using Adobe Bridge and Device Central, see http://www.adobe.com/go/vid0208.

Working with projects

About projects

You can use Flash projects (FLP files) to manage multiple document files in a single project. Flash projects allow you to group multiple, related files together to create complex applications.

You can use version-control features with projects to ensure that the correct file versions are used during editing, and to prevent accidental overwriting.

Flash projects include the following features:

- A Flash project can contain any Flash or other file type, including previous versions of FLA and SWF files.
- You can add an existing file to a Flash project. Each file can be added to a particular Flash project only once. Files can be organized in nested folders.
- A Flash project is an XML file with the file extension .flp—for example, myProject.flp. The XML file references all the document files contained in the Flash project.
- A Flash project can contain another Flash project (FLP file).
- Changes that you make to a project are updated to the FLP file immediately, so the file is always current; you do not need to do save the file.
- You can create a Flash project in the Flash authoring environment, or you can create the XML file for a Flash project in an external application.
- Flash projects use UTF-8 text encoding. All filenames and folder names in a Flash project must be UTF-8 compatible.

Create and manage projects

You use the Project panel (Window > Project) to create and manage projects. The panel displays the contents of a Flash project in a collapsible tree structure. The panel title bar displays the project name.

If a project file is missing (not in its specified location), a Missing File icon appears next to the filename. You can search for the missing file or delete the file from the project.

When you publish a project, each FLA file in the project is published with the publish profile specified for that file.

Only one project can be open at one time. If a project is open and you open or create another project, Flash automatically saves and closes the first file.

See also

“Using publish profiles” on page 434

View the Project pop-up menu

❖ When a project is open, click the Project button at the upper-left corner of the Project panel.

Create a new project

1 Do one of the following:

- Select File > New. On the General tab, select Flash Project.
- Select New Project from the Project pop-up menu in the Project panel (visible only if a project is open).

- If no other project is open, open the Project panel and select Create A New Project in the panel window.
- If no project is currently open, right-click (Windows) or Control-click (Macintosh) in the Document window of a saved Flash document or ActionScript file and select Add To New Project from the context menu.

2 In the New Project dialog box, enter a name for the project and click Save.

Open an existing project

❖ Do one of the following:

- Select Open Project from the Project pop-up menu in the Project panel. Navigate to the project and click Open.
- Double-click the filename.
- If no other project is open, open the Project panel and select Open An Existing Project in the panel window. Navigate to the project and click Open.
- Select File > Open. Navigate to the project and click Open.

Add a file to a project

❖ Do one of the following:

- Click the Add Files (+) button at the lower-right corner of the Project panel. Select one or more files and click Add.
- Right-click (Windows) or Control-click (Macintosh) in the Document window of an open FLA or AS file and select Add To Project from the context menu.

Note: *A file must be saved before you can add it to a project. You can add a file to a given project only once. If you attempt to add a file to the same project more than once, an error message appears.*

Create a folder

1 Click the Folder button at the lower-right corner of the Project panel.

2 Enter a name for the folder and click OK.

Note: *Folders at the same level on the same branch of the project tree structure must have unique names. If there is a folder name conflict, an error message appears.*

Move a file or folder

❖ Drag the file or folder to a new location in the project tree structure. When you move a folder, all of its contents are moved.

Note: *If you drag a folder to a location with another folder of the same name, Flash merges the contents of the two folders in the new location.*

Delete a file or folder

1 Select the item in the Project panel.

2 Do one of the following:

- Click the Remove button at the lower-right corner of the Project panel.
- Press the Delete key.
- Right-click (Windows) or Control-click (Macintosh) the file or folder and select Remove from the context menu.

Open a file from the Project panel in Flash

❖ Double-click the filename in the Project panel.

If the file is of a native file type (a type supported by the Flash authoring tool), the file opens in Flash. If the file is a nonnative file type, the file opens in the application used to create it.

Test a project

1 Click Test Project in the Project panel.

2 If the project contains no FLA, HTML, or HTM file, an error message appears. Click OK and add a file of the appropriate type.

3 If no FLA, HTML, or HTM file is designated as the default document, an error message appears. Click OK. In the Select Default Document dialog box, select a document and click OK.

When a default document is present, the Test Project feature publishes all FLA files in the document. If the default document is a FLA file, the Test Movie command is executed. If it is an HTML file, a browser is opened.

Specify a publish profile for a FLA file in a project

1 Select the file in the Project panel and do one of the following:

- Select Settings from the Project pop-up menu.
- Right-click (Windows) or Control-click (Macintosh) and select Settings from the context menu.

2 In the Project Settings dialog box, select the FLA file in the tree structure.

3 Select a publish profile from the Profile menu.

Publish a project

❖ Select Publish Project from the Project pop-up menu.

***Note:** Flash uses default publish profiles to publish FLA files in the project, unless you select other profiles.*

Save files in a project when testing or publishing

1 Select Edit > Preferences (Windows) or Flash > Preferences (Macintosh) and click General in the Category list.

2 Under Project Preferences, click Save Files On Test or Publish Project.

When this option is selected, Flash saves all open files in the current project before executing the Test Project or Publish Project operation.

Close a project

❖ Select Close Project from the Project pop-up menu.

Set preferences to close all files or not close all files when you close a project

1 Select Edit > Preferences (Windows) or Flash > Preferences (Macintosh) and click General in the Category list.

2 Under Project Preferences, select or deselect Close Files with Project. When this option is selected, by default, Flash closes all files in a project when you close the project.

Rename a project or a folder

1 Select the project name or folder name in the Project panel and do one of the following:

- Select Rename from the Project pop-up menu.
- Right-click (Windows) or Control-click (Macintosh) the item and select Rename from the context menu.

2 Enter a new name and click OK.

***Note:** By default, a project is given the same name as the first file added to the project. To rename a project, you must use the Rename menu item. Renaming the FLP file for a project does not rename the project.*

Find a missing file in a project

A file that is part of a project can appear to be missing if it is moved from its original location relative to the other files in the project.

1 Select the filename (designated by a Missing File icon) in the Project panel.

2 Select Find Missing File from the Project pop-up menu, or right-click (Windows) or Control-click (Macintosh) and select Find Missing File from the context menu.

3 Navigate to the file and click OK.

Use version control with projects

Version control lets you ensure that each author working in a project file is always using the latest version of a file, and that multiple authors do not overwrite each other’s work.

To use version-control features, you must define a site for the project. You can specify a local, network, or FTP connection, or you can specify custom plug-ins for version-control systems.

On Windows, you can use Flash projects with SourceSafe.

Define a site for version control

1 Create a new project and add files.

2 Select File > Edit Sites.

3 In the Edit Sites dialog box, click New.

4 In the Site Definition dialog box, enter the site name, the local root path, and the e-mail address and name of the user.

5 To specify a local, network, or FTP connection, select Local/Network or FTP from the Connection menu. Enter the location information for the Local/Network path or for the FTP connection and skip the next step.

6 (Windows only) To specify a Visual SourceSafe database, select SourceSafe Database from the Connection menu.

a In the Database Path box, click Browse to browse for the VSS database you want, or enter the full file path. The file you select becomes the srcsafe.ini file used to initialize SourceSafe.

b In the Project box, enter the project within the VSS database that you want to use as the remote site’s root directory.

c In the Username and Password boxes, enter your login user name and password for the selected database. If you don’t know your user name and password, check with your system administrator.

d Click OK to return to the Site Definition dialog box.

***Note:** You must have Microsoft Visual SourceSafe Client version 6 installed.*

7 In the Project panel (Window > Project), select Settings from the Project pop-up menu or context menu.

8 In the Project Settings dialog box, select the site definition from the Site menu in the Version Control section. Click OK.

9 In the Project pop-up menu, select Check In. Flash checks all files in the current project into the site.

Edit a file with version control applied

1 Open the project that contains the file.

2 Select the file in the tree structure in the project panel and select Check Out from the project context menu.

The icon next to the filename indicates that the file is checked out.

3 To check a file back in, select the file in the project panel and select Check In from the project context menu.

The icon next to the filename indicates that the file is checked in.

Open a file from a version-control site

1 Select File > Open from Site.

2 In the Open From Site dialog box, select the site from the Site menu.

3 Select the file in the site.

4 If the file exists on your local system, a message appears indicating whether the file is checked out and, if so, asking whether you want to overwrite it. Click Yes to overwrite the local version with the version from the remote site.

Troubleshooting remote folder setup for Flash projects

A web server can be configured in many ways. The following information can help you resolve some common issues in setting up a remote folder for version control:

- The Flash FTP implementation may not work properly with certain proxy servers, multilevel firewalls, and other forms of indirect server access. If you have problems with FTP access, ask your local system administrator for help.
- In some applications, you can connect to any remote directory, then navigate through the remote file system to find the directory you want. However, for the Flash FTP implementation, you must connect to the remote system's root folder. Be sure to indicate the remote system's root folder as the host directory.
- If you have problems connecting, and you've specified the host directory using a single slash (/), you might need to specify a relative path between the directory you are connecting to and the remote root folder. For example, if the remote root folder is a higher-level directory, you may need to specify a ../../ for the host directory.
- Filenames and folder names that contain spaces and special characters often cause problems when transferred to a remote site. Use underscores in place of spaces, and avoid special characters wherever possible, especially colons, slashes, periods, and apostrophes.
- If problems persist, try uploading with an external FTP program to determine if the problem is specific to using FTP in Flash.

Adding media to the library

Managing media assets with the Flash document library

The library in a Flash document stores media assets that you create in the Flash authoring environment or import to use in the document. You can create vector artwork or text directly in Flash; import vector artwork, bitmaps, video, and sound; and create symbols. A *symbol* is a graphic, a button, a movie clip, or text that you create once and can reuse multiple times. You can also use ActionScript to add media content to a document dynamically.

The library also contains any components that you have added to your document. Components appear in the library as compiled clips.

You can open the library of any Flash document while you are working in Flash, to make the library items from that file available for the current document.

You can create permanent libraries in your Flash application that are available whenever you start Flash. Flash also includes several sample libraries containing buttons, graphics, movie clips, and sounds.

You can export library assets as a SWF file to a URL to create a runtime-shared library. This lets you link to the library assets from Flash documents that import symbols using runtime sharing.

See also

"Working with text" on page 260

"Using imported artwork" on page 132

"Working with sound" on page 291

"Working with video" on page 300

"Using symbols, instances, and library assets" on page 207

Work with libraries

The Library panel (Window > Library) displays a scroll list with the names of all items in the library, which lets you view and organize these elements as you work. An icon next to an item's name in the Library panel indicates the item's file type.

Open a library in another Flash file

1 From the current document, select File > Import > Open External Library.

2 Navigate to the Flash file whose library you want to open and click Open.

The selected file's library opens in the current document, with the filename at the top of the Library panel. To use items from the selected file's library in the current document, drag the items to the current document's Library panel or to the Stage.

Resize the Library panel

❖ Do one of the following:

- Drag the lower-right corner of the panel.
- Click the Wide State button to enlarge the Library panel so it shows all the columns.
- Click the Narrow State button to reduce the width of the Library panel.

Change the width of columns

❖ Position the pointer between column headers and drag to resize.

You cannot change the order of columns.

Access the Panel menu for the Library panel

❖ Click the Panel menu button in the Library panel's title bar.

Work with library items

When you select an item in the Library panel, a thumbnail preview of the item appears at the top of the Library panel. If the selected item is animated or is a sound file, you can use the Play button in the library preview window or the Controller to preview the item.

Use a library item in the current document

❖ Drag the item from the Library panel onto the Stage.

The item is added to the current layer.

Convert an object on the Stage to a symbol in the library

❖ Drag the item from the Stage onto the current Library panel.

Use a library item from the current document in another document

❖ Drag the item from the Library panel or Stage into the Library panel or Stage of another document.

Copy library items from a different document

1 Select the document that contains the library items.

2 Select the library items in the Library panel.

3 Select Edit > Copy.

4 Select the document that you want to copy the library items to.

5 Select that document’s Library panel.

6 Select Edit > Paste.

Work with folders in the Library panel

You can organize items in the Library panel using folders. When you create a new symbol, it is stored in the selected folder. If no folder is selected, the symbol is stored at the root of the library.

Create a new folder

❖ Click the New Folder button at the bottom of the Library panel.

Open or close a folder

❖ Double-click the folder, or Select the folder and select Expand Folder or Collapse Folder from the Panel menu for the Library panel.

Open or close all folders

❖ Select Expand All Folders or Collapse All Folders from the Panel menu for the Library panel.

Move an item between folders

❖ Drag the item from one folder to another.

If an item with the same name exists in the new location, Flash prompts you to replace it with the item you are moving.

Sort items in the Library panel

Columns in the Library panel list the name of an item, its type, the number of times it’s used in the file, its linkage status and identifier (if the item is associated with a shared library or is exported for ActionScript), and the date on which it was last modified.

You can sort items in the Library panel alphanumerically by any column. Items are sorted within folders.

❖ Click the column header to sort by that column. Click the triangle button to the right of the column headers to reverse the sort order.

Edit a library item

1 Select the item in the Library panel.

2 Select one of the following from the Panel menu for the Library panel:

- To edit an item in Flash, select Edit.
- To edit an item in another application, select Edit With and then select an external application.

Note: *When starting a supported external editor, Flash opens the original imported document.*

Rename a library item

Changing the library item name of an imported file does not change the filename.

1 Do one of the following:

- Double-click the item's name.
- Select the item and select Rename from the Panel menu for the Library panel.
- Right-click (Windows) or Control-click (Macintosh) the item and select Rename from the context menu.

2 Enter the new name in the box.

Delete a library item

When you delete an item from the library, all instances or occurrences of that item in the document are also deleted unless you specify that they not be.

1 Select the item and click the trash can icon at the bottom of the Library panel.

2 In the warning box that appears, select Delete Symbol Instances (the default) to delete the library item and all its instances. Deselect the option to delete only the symbol, which leaves the instances on the Stage.

3 Click Delete.

Find unused library items

To organize your document, you can find unused library items and delete them.

Note: *It is not necessary to delete unused library items to reduce a Flash document's file size, because unused library items are not included in the SWF file. However, items linked for export are included in the SWF file.*

❖ Do one of the following:

- Select Unused Items from the Panel menu for the Library panel.
- Sort library items by the Use Count column, which indicates whether an item is in use.

See also

"Using shared library assets" on page 218

Update imported files in the library

If you use an external editor to modify files that you have imported into Flash, such as bitmaps or sound files, you can update the files in Flash without reimporting them. You can also update symbols that you have imported from external Flash documents. Updating an imported file replaces its contents with the contents of the external file.

1 Select the imported file in the Library panel.

2 Select Update from the Panel menu for the Library panel.

Work with common libraries

You can use the sample common libraries included with Flash to add buttons or sounds to your documents. You can also create custom common libraries, which you can then use with any documents that you create.

See also

“Configuration folders installed with Flash” on page 416

Use an item from a common library in a document

1 Select Window > Common Libraries, and select a library from the submenu.

2 Drag an item from the common library into the library for the current document.

Create a common library for your Flash application

1 Create a Flash file with a library containing the symbols that you want to include in the common library.

2 Place the Flash file in the user-level Libraries folder on your hard disk.

- On Windows, the path is C:\Documents and Settings\username\Local Settings\Application Data\Adobe\Flash CS3\language\Configuration\Libraries\.
- On Mac OS, the path is Hard Disk/Users/*username*/Library/Application Support/Adobe/Flash CS3/*language*/Configuration/Libraries/.

Working with timelines

About frames and keyframes

Like films, Flash documents divide lengths of time into frames. In the Timeline, you work with these frames to organize and control your document’s content. You place frames in the Timeline in the order you want the objects in the frames to appear in your finished content.

A *keyframe* is a frame in which you define a change to an object’s properties for an animation or include ActionScript code to control some aspect of your document. You can also arrange keyframes in the Timeline to edit the sequence of events in an animation. Flash can *tween*, or automatically fill in, the frames between keyframes in order to produce fluid animations. Because keyframes let you produce animation without drawing each individual frame, they make creating animation easier.

For a video tutorial about the Timeline, keyframes, and frame rates, see www.adobe.com/go/vid0123.

See also

"Creating animation" on page 228

"The Timeline" on page 33

Manage frames and keyframes in the Timeline

You can perform the following modifications on frames or keyframes:

- Insert, select, delete, and move frames or keyframes.
- Drag frames and keyframes to a new location on the same layer or on a different layer.
- Copy and paste frames and keyframes.
- Convert keyframes to frames.
- Drag an item from the Library panel onto the Stage to add the item to the current keyframe.

Flash offers two different methods for selecting frames in the Timeline. In frame-based selection (the default) you select individual frames in the Timeline. In span-based selection, the entire frame sequence, from one keyframe to the next, is selected when you click any frame in the sequence. You can specify span-based selection in Flash preferences.

Specify span-based frame selection

1. Select Edit > Preferences.
2. Select the General category.
3. In the Timeline section, select Span Based Selection.
4. Click OK.

Insert frames in the Timeline

- To insert a new frame, select Insert > Frame.
- To create a new keyframe, select Insert > Keyframe, or right-click (Windows) or Control-click (Macintosh) the frame where you want to place a keyframe, and select Insert Keyframe from the context menu.
- To create a new blank keyframe, select Insert > Blank Keyframe, or right-click (Windows) or Control-click (Macintosh) the frame where you want to place the keyframe, and select Insert Blank Keyframe from the context menu.

Select one or more frames in the Timeline

- To select one frame, click the frame. If you have Span Based Selection enabled, clicking one frame selects the entire frame sequence between two keyframes.
- To select multiple contiguous frames, Shift-click additional frames.
- To select multiple discontiguous frames, Control-click (Windows) or Command-click (Macintosh) additional frames.
- To select all frames in the Timeline, select Edit > Timeline > Select All Frames.

Copy or paste a frame or frame sequence

❖ Do one of the following:

- Select the frame or sequence and select Edit > Timeline > Copy Frames. Select the frame or sequence that you want to replace, and select Edit > Timeline > Paste Frames.
- Alt-click (Windows) or Option-click (Macintosh) and drag the keyframe to the location where you want to paste it.

Delete a frame or frame sequence

❖ Select the frame or sequence and select Edit > Timeline > Remove Frame, or right-click (Windows) or Control-click (Macintosh) the frame or sequence and select Remove Frame from the context menu.

Surrounding frames remain unchanged.

Move a keyframe or frame sequence and its contents

❖ Drag the keyframe or sequence to the desired location.

Change the length of a tweened sequence

❖ Drag the beginning or ending keyframe left or right. To change the length of a frame-by-frame animation sequence, see “Create frame-by-frame animations” on page 230.

Extend the duration of a keyframe animation

❖ Press Alt (Windows) or Option (Macintosh) and drag the keyframe to the frame that you want to be the final frame of the sequence.

Convert a keyframe to a frame

❖ Select the keyframe and select Edit > Timeline > Clear Keyframe, or right-click (Windows) or Control-click (Macintosh) the keyframe and select Clear Keyframe from the context menu.

The Stage contents of the cleared keyframe and all frames up to the subsequent keyframe are replaced with the Stage contents of the frame preceding the cleared keyframe.

Add an item from the library to the current keyframe

❖ Drag the item from the Library panel onto the Stage.

About multiple timelines and levels

Flash Player has a stacking order of levels. Every Flash document has a main Timeline located at level 0 in Flash Player. You can use the `loadMovie` action to load other Flash documents (SWF files) into Flash Player at different levels.

If you load documents into levels above level 0, the documents stack on top of one another like drawings on transparent paper; when there is no content on the Stage, you can see through to the content on lower levels. If you load a document into level 0, it replaces the main Timeline. Each document loaded into a level of Flash Player has its own Timeline.

Timelines can send messages to each other with ActionScript. For example, an action on the last frame of one movie clip can tell another movie clip to play. To use ActionScript to control a Timeline, you must use a target path to specify the location of the Timeline.

For a video tutorial about using multiple timelines, see www.adobe.com/go/vid0128.

About nested movie clips and parent-child hierarchy

When you create a movie clip instance in a Flash document, the movie clip's Timeline is nested inside the main Timeline of the document. Each movie clip symbol has its own Timeline. You can also nest a movie clip instance inside another movie clip symbol.

When a movie clip is nested inside another movie clip, or inside a document, it becomes a child of that movie clip or document, which becomes the parent. Relationships between nested movie clips are hierarchical: modifications made to the parent affect the child. The root Timeline for each level is the parent of all the movie clips on its level, and because it is the topmost Timeline, it has no parent. In the Movie Explorer, you can view the hierarchy of nested movie clips in a document.

To understand movie clip hierarchy, consider the hierarchy on a computer: the hard disk has a root directory (or folder) and subdirectories. The root directory is analogous to the main (or root) Timeline of a Flash document: it is the parent of everything else. The subdirectories are analogous to movie clips.

You can use the movie clip hierarchy in Flash to organize related objects. For example, you could create a Flash document containing a car that moves across the Stage. You can use a movie clip symbol to represent the car and set up a motion tween to move it across the Stage.

To add wheels that rotate, you can create a movie clip for a car wheel, and create two instances of this movie clip, named `frontWheel` and `backWheel`. Then you can place the wheels on the car movie clip's Timeline—not on the main Timeline. As children of `car`, `frontWheel` and `backWheel` are affected by any changes made to `car`; they move with the car as it tweens across the Stage.

To make both wheel instances spin, you can set up a motion tween that rotates the wheel symbol. Even after you change `frontWheel` and `backWheel`, they continue to be affected by the tween on their parent movie clip, `car`; the wheels spin, but they also move with the parent movie clip `car` across the Stage.

For a video tutorial about using multiple timelines, see www.adobe.com/go/vid0128.

See also

"Using symbols, instances, and library assets" on page 207

About absolute paths

An absolute path starts with the name of the level into which the document is loaded and continues through the display list until it reaches the target instance. You can also use the alias `_root` to refer to the topmost Timeline of the current level. For example, an action in the movie clip `california` that refers to the movie clip `oregon` could use the absolute path `_root.westCoast.oregon`.

The first document to open in Flash Player is loaded at level 0. You must assign each additional loaded document a level number. When you use an absolute reference in ActionScript to reference a loaded document, use the form `_levelX`, where *`X`* is the level number into which the document is loaded. For example, the first document that opens in Flash Player is called `_level0`; a document loaded into level 3 is called `_level3`.

To communicate between documents on different levels, you must use the level name in the target path. The following example shows how the `portland` instance would address the `atlanta` instance located in a movie clip called `georgia` (`georgia` is at the same level as `oregon`):

```
_level5.georgia.atlanta
```

You can use the `_root` alias to refer to the main Timeline of the current level. For the main Timeline, the `_root` alias stands for `_level0` when targeted by a movie clip also on `_level0`. For a document loaded into `_level5`, `_root` is equal to `_level5` when targeted by a movie clip also on level 5. For example, if the movie clips `southcarolina` and `florida` are both loaded into the same level, an action called from the instance `southcarolina` could use the following absolute path to target the instance `florida`:

```
_root.eastCoast.florida
```

About relative paths

A relative path depends on the relationship between the controlling Timeline and the target Timeline. Relative paths can address targets only within their own level of Flash Player. For example, you can't use a relative path in an action on `_level0` that targets a Timeline on `_level5`.

In a relative path, use the keyword `this` to refer to the current Timeline in the current level; use the `_parent` alias to indicate the parent Timeline of the current Timeline. You can use the `_parent` alias repeatedly to go up one level in the movie clip hierarchy within the same level of Flash Player. For example, `_parent._parent` controls a movie clip up two levels in the hierarchy. The topmost Timeline at any level in Flash Player is the only Timeline with a `_parent` value that is undefined.

An action in the Timeline of the instance `charleston`, located one level below `southcarolina`, could use the following target path to target the instance `southcarolina`:

```
_parent
```

To target the instance `eastCoast` (one level up) from an action in `charleston`, you could use the following relative path:

```
_parent._parent
```

To target the instance `atlanta` from an action in the Timeline of `charleston`, you could use the following relative path:

```
_parent._parent.georgia.atlanta
```

Relative paths are useful for reusing scripts. For example, you could attach the following script to a movie clip that magnifies its parent by 150%:

```
onClipEvent (load) {_parent._xscale = 150;_parent._yscale = 150;
}
```

You can reuse this script by attaching it to any movie clip instance.

***Note:** Flash Lite 1.0 and 1.1 support attaching scripts only to buttons. Attaching scripts to movie clips is not supported.*

Whether you use an absolute or a relative path, you identify a variable in a Timeline or a property of an object with a dot (.) followed by the name of the variable or property. For example, the following statement sets the variable `name` in the instance `form` to the value `"Gilbert"`:

```
_root.form.name = "Gilbert";
```

Using absolute and relative target paths

You can use ActionScript to send messages from one timeline to another. The timeline that contains the action is called the *controlling timeline*, and the timeline that receives the action is called the *target timeline*. For example, there could be an action on the last frame of one timeline that tells another timeline to play. To refer to a target timeline, you must use a target path, which indicates the location of a movie clip in the display list.

The following example shows the hierarchy of a document named westCoast on level 0, which contains three movie clips: `california`, `oregon`, and `washington`. Each of these movie clips in turn contains two movie clips.

```
_level0
        westCoast
                california
                        sanfrancisco
                        bakersfield
                oregon
                        portland
                        ashland
                washington
                        olympia
                        ellensburg
```

As on a web server, each timeline in Flash can be addressed in two ways: with an absolute path or with a relative path. The absolute path of an instance is always a full path from a level name, regardless of which timeline calls the action; for example, the absolute path to the instance `california` is `_level0.westCoast.california`. A relative path is different when called from different locations; for example, the relative path to `california` from `sanfrancisco` is `_parent`, but from `portland`, it's `_parent._parent.california`.

See also

"Structuring FLA files" on page 464

"Organizing ActionScript in an application" on page 466

Specify target paths

To control a movie clip, loaded SWF file, or button, you must specify a target path. You can specify it manually, or by using the Insert Target Path dialog box, or by creating an expression that evaluates to a target path. To specify a target path for a movie clip or button, you must assign an instance name to the movie clip or button. A loaded document doesn't require an instance name, because you use its level number as an instance name (for example, `_level5`).

Assign an instance name to a movie clip or button

1 Select a movie clip or button on the Stage.

2 Enter an instance name in the Property inspector.

Specify a target path using the Insert Target Path dialog box

1 Select the movie clip, frame, or button instance to which you want to assign the action.

This becomes the controlling Timeline.

2 In the Actions panel (Window > Actions), go to the Actions toolbox on the left, and select an action or method that requires a target path.

3 Click the parameter box or location in the script where you want to insert the target path.

4 Click the Insert Target Path button ⊕ above the Script pane.

5 Select Absolute or Relative for the target path mode.

6 Select a movie clip in the Insert Target Path display list, and click OK.

Specify a target path manually

1 Select the movie clip, frame, or button instance to which you want to assign the action.

This becomes the controlling Timeline.

2 In the Actions panel (Window > Actions), go to the Actions toolbox on the left, and select an action or method that requires a target path.

3 Click the parameter box or location in the script where you want to insert the target path.

4 Enter an absolute or relative target path in the Actions panel.

Use an expression as a target path

1 Select the movie clip, frame, or button instance to which you want to assign the action.

This becomes the controlling Timeline.

2 In the Actions panel (Window > Actions), go to the Actions toolbox on the left, and select an action or method that requires a target path.

3 Do one of the following:

- Enter an expression that evaluates to a target path in a parameter box.
- Click to place the insertion point in the script. Then, in the Functions category of the Actions toolbox, double-click the `targetPath` function. The `targetPath` function converts a reference to a movie clip into a string.
- Click to place the insertion point in the script. Then, in the Functions category of the Actions toolbox, select the `eval` function. The `eval` function converts a string to a movie clip reference that can be used to call methods such as `play`.

The following script assigns the value 1 to the variable `i`. It then uses the `eval` function to create a reference to a movie clip instance and assigns it to the variable `x`. The variable `x` is now a reference to a movie clip instance and can call the MovieClip object methods.

```
0i = 1;
0x = eval("mc"+i);
0x.play();
0// this is equivalent to mc1.play();
```

You can also use the `eval` function to call methods directly, as shown in the following example:

```
0eval("mc" + i).play();
```

Working with scenes

About scenes

To organize a document thematically, you can use scenes. For example, you might use separate scenes for an introduction, a loading message, and credits. Though using scenes has some disadvantages, there are some situations in which few of these disadvantages apply, such as when you create lengthy animations. When you use scenes, you avoid having to manage a large number of FLA files.

Using scenes is similar to using several SWF files together to create a larger presentation. Each scene has a Timeline. When the playhead reaches the final frame of a scene, the playhead progresses to the next scene. When you publish a SWF file, the Timeline of each scene combines into a single Timeline in the SWF file. After the SWF file compiles, it behaves as if you created the FLA file using one scene. Because of this behavior, scenes have some disadvantages:

- Scenes can make documents confusing to edit, particularly in multiauthor environments. Anyone using the FLA document might have to search several scenes within a FLA file to locate code and assets. Consider loading content or using movie clips instead.
- Scenes often result in large SWF files. Using scenes encourages you to place more content in a single FLA file, which results in larger FLA files and SWF files.
- Scenes force users to progressively download the entire SWF file, even if they do not plan or want to watch all of it. If you avoid scenes, users can control what content they download as they progress through your SWF file.
- Scenes combined with ActionScript might produce unexpected results. Because each scene Timeline is compressed onto a single Timeline, you might encounter errors involving your ActionScript and scenes, which typically requires extra, complicated debugging.

Use scenes

When you publish a Flash document that contains more than one scene, the scenes in the document play back in the order they are listed in the Scene panel. Frames in the document are numbered consecutively through scenes. For example, if a document contains two scenes with ten frames each, the frames in Scene 2 are numbered 11–20.

To stop or pause a document after each scene, or to let users navigate the document in a nonlinear fashion, you use actions.

Display the Scene panel

❖ Select Window > Other Panels > Scene.

View a particular scene

❖ Select View > Go To, and then select the name of the scene from the submenu.

Add a scene

❖ Select Insert > Scene, or click the Add Scene button ✚ in the Scene panel.

Delete a scene

❖ Click the Delete Scene button 🗑 in the Scene panel.

Change the name of a scene

❖ Double-click the scene name in the Scene panel and enter the new name.

Duplicate a scene

❖ Click the Duplicate Scene button ⧉ in the Scene panel.

Change the order of a scene in the document

❖ Drag the scene name to a different location in the Scene panel.

Find and Replace

About Find and Replace

The Find and Replace feature lets you do the following:

- Search for a text string, a font, a color, a symbol, a sound file, a video file, or an imported bitmap file.
- Replace the specified element with another element of the same type. Different options are available in the Find and Replace dialog box depending on the type of specified element.
- Find and replace elements in the current document or the current scene.
- Search for the next occurrence or all occurrences of an element, and replace the current occurrence or all occurrences.

***Note:** In a screen-based document, you can find and replace elements in the current document or the current screen, but you can't use scenes.*

The Live Edit option lets you edit the specified element directly on the Stage. If you use Live Edit when searching for a symbol, Flash opens the symbol in edit-in-place mode.

The Find and Replace Log at the bottom of the Find and Replace dialog box shows the location, name, and type of the elements for which you are searching.

See also

"Working with screens" on page 366

Find and replace text

1 Select Edit > Find and Replace.

2 Select Text from the For pop-up menu.

3 In the Text box, enter the text to find.

4 In the Replace With Text box, enter the text to replace the existing text.

5 Select options for searching text:

Whole Word Searches for the specified text string as a whole word only, bounded on both sides by spaces, quotes, or similar markers. When Whole Word is deselected, the specified text can be searched as part of a larger word. For example, when Whole Word is deselected, a search for *place* will yield the words *replace*, *placement*, and so on.

Match Case Searches for text that exactly matches the case (uppercase and lowercase character formatting) of the specified text when finding and replacing.

Regular Expressions Searches for text in regular expressions in ActionScript. An expression is any statement that Flash can evaluate that returns a value.

Text Field Contents Searches the contents of a text field.

Frames/Layers/Parameters Searches frame labels, layer names, scene names, and component parameters.

Strings in ActionScript Searches strings in ActionScript in the document or scene (external ActionScript files are not searched).

6 To select the next occurrence of the specified text on the Stage and edit it in place, select Live Edit.

***Note:** Only the next occurrence is selected for live editing, even if you select Find All in step 7.*

7 To find text, do one of the following:

- To find the next occurrence of the specified text, click Find Next.
- To find all occurrences of the specified text, click Find All.

8 To replace text, do one of the following:

- To replace the currently selected occurrence of the specified text, click Replace.
- To replace all occurrences of the specified text, click Replace All.

Find and replace fonts

1 Select Edit > Find And Replace.

2 Select Font from the For pop-up menu, then select from the following options:

- To search by font name, select Font Name and select a font from the pop-up menu or enter a font name in the box. When Font Name is deselected, all fonts in the scene or document are searched.
- To search by font style, select Font Style and select a font style from the pop-up menu. When Font Style is deselected, all font styles in the scene or document are searched.
- To search by font size, select Font Size and enter a value for minimum and maximum font size to specify the range of font sizes to be searched. When Font Size is deselected, all font sizes in the scene or document are searched.
- To replace the specified font with a different font name, select Font Name under Replace With and select a font name from the pop-up menu or enter a name in the box. When Font Name is deselected under Replace with, the current font name remains unchanged.
- To replace the specified font with a different font style, select Font Style under Replace With and select a font style from the pop-up menu. When Font Style is deselected under Replace with, the current style of the specified font remains unchanged.
- To replace the specified font with a different font size, select Font Size under Replace With and enter values for minimum and maximum font size. When Font Size is deselected under Replace With, the current size of the specified font remains unchanged.

3 To select the next occurrence of the specified font on the Stage and edit it in place, select Live Edit.

***Note:** Only the next occurrence is selected for live editing, even if you select Find All in step 4.*

4 To find a font, do one of the following:

- To find the next occurrence of the specified font, click Find Next.
- To find all occurrences of the specified font, click Find All.

5 To replace a font, do one of the following:

- To replace the currently selected occurrence of the specified font, click Replace.
- To replace all occurrences of the specified font, click Replace All.

Find and replace colors

You cannot find and replace colors in grouped objects.

***Note:** To find and replace colors in a GIF or JPEG file in a Flash document, edit the file in an image-editing application.*

1 Select Edit > Find And Replace.

2 Select Color from the For pop-up menu.

3 To search for a color, click the Color control and do one of the following:

- Select a color swatch from the color pop-up window.
- Enter a hexadecimal color value in the Hex Edit box in the color pop-up window.
- Click the System Color Picker button and select a color from the system color picker.
- To make the eyedropper tool appear, drag from the Color control. Select any color on your screen.

4 To select a color to replace the specified color, click the Color control under Replace With and do one of the following:

- Select a color swatch from the color pop-up window.
- Enter a hexadecimal color value in the Hex Edit box in the color pop-up window.
- Click the System Color Picker button and select a color from the system color picker.
- To make the eyedropper tool appear, drag from the Color control. Select any color on your screen.

5 To specify which occurrence of the color to find and replace, select the Fills, Strokes, or Text option or any combination of those options.

6 To select the next occurrence of the specified color on the Stage and edit it in place, select Live Edit.

Note: *Only the next occurrence is selected for live editing, even if you select Find All in the next step.*

7 To find a color, do one of the following:

- To find the next occurrence of the specified color, click Find Next.
- To find all occurrences of the specified color, Click Find All.

8 To replace a color, do one of the following:

- To replace the currently selected occurrence of the specified color, click Replace.
- To replace all occurrences of the specified color, click Replace All.

Find and replace symbols

When you find and replace symbols, search for a symbol by name. Replace a symbol with another symbol of any type—movie clip, button, or graphic.

1 Select Edit > Find And Replace.

2 Select Symbol from the For pop-up menu.

3 For Name, select a name from the pop-up menu.

4 Under Replace With, for Name select a name from the pop-up menu.

5 To select the next occurrence of the specified symbol on the Stage and edit it in place, select Live Edit.

Note: *Only the next occurrence is selected for editing, even if you select Find All in the next step.*

6 To find a symbol, do one of the following:

- To find the next occurrence of the specified symbol, click Find Next.
- To find all occurrences of the specified symbol, click Find All.

7 To replace a symbol, do one of the following:

- To replace the currently selected occurrence of the specified symbol, click Replace.
- To replace all occurrences of the specified symbol, click Replace All.

Find and replace sound, video, or bitmap files

1 Select Edit > Find and Replace.

2 Select Sound, Video, or Bitmap from the For pop-up menu.

3 For Name, enter a sound, video, or bitmap filename or select a name from the pop-up menu.

4 Under Replace With, for Name enter a sound, video, or bitmap filename or select a name from the pop-up menu.

5 To select the next occurrence of the specified sound, video, or bitmap on the Stage and edit it in place, select Live Edit.

Note: *Only the next occurrence is selected for editing, even if you select Find All in the next step.*

6 To find a sound, video, or bitmap, do one of the following:

- To find the next occurrence of the specified sound, video, or bitmap, click Find Next.
- To find all occurrences of the specified sound, video, or bitmap, click Find All.

7 To replace a sound, video, or bitmap, do one of the following:

- To replace the currently selected occurrence of the specified sound, video, or bitmap, Click Replace.
- To replace all occurrences of the specified sound, video, or bitmap, click Replace All.

Templates

About templates

The Flash templates provide you with easy-to-use starting points for a range of common projects. There are templates for projects such as photo slideshows, quizzes, mobile content, and more.

Mobile device templates

Adobe® Flash® CS3 Professional content is viewable across multiple browsers, platforms, and mobile phones. You can author the following:

- High-quality animations
- Games
- Rich-media custom user interfaces for devices and desktop systems
- E-commerce and business solutions

Flash files are compact, which makes them suitable for wireless carrier networks, where transfer rates vary between 9.6 and 60 kilobytes per second (Kbps). Mobile devices have limited storage capability, so the small memory requirement of Flash is ideal.

Mobile device templates let you create content for many mobile devices. Use the device skins in the templates to preview content as it will look on the device.

Note: *The skins are on guide layers and won’t export with your content or appear at runtime.*

For more information on authoring Flash files for mobile devices, see the Adobe Mobile Devices site at www.adobe.com/go/devnet_devices.

Quiz templates

Use the quiz templates to create self-scoring quizzes with several interaction types.

Advertising templates

Advertising templates facilitate the creation of standard rich media types and sizes defined by the Interactive Advertising Bureau (IAB) and accepted by the industry. For more information on IAB-endorsed ad types, see the IAB site at IAB.net.

Test ads for stability in a variety of browser and platform combinations. Your application is considered stable if it doesn't cause error messages, browser crashes, or system crashes.

Work with webmasters and network administrators to create detailed testing plans that include tasks relevant to your users. Make these plans publicly available and update them regularly. Vendors should publish detailed plans indicating the browser and platform combinations in which their technologies are stable. Examples are available at the IAB Rich Media testing section of IAB.net. Size and file format requirements of ads might vary by vendor and site. Check with your vendor, ISP, or the IAB to learn about these requirements that can affect the ad's design.

Photo slideshow templates

Use the photo slideshow template to exhibit your photos with text captions and playback controls.

To use templates

1. Select File > New.
2. Click the Templates tab.
3. Select a template and click OK.
4. Add content to the FLA file.
5. Save and publish the file.

Use the Photo Slideshow template

Photos must be in a suitable format for use in the Photo Slideshow template. You can import images in a variety of formats, but JPEGs typically work best for photographs. For best results, save photos as JPEGs using an image-editing program. Images should be 640 x 480 pixels and should be named in a numbered sequence, for example, photo1.jpg, photo2.jpg, and photo3.jpg.

Import photos to a SWF file

1. Select the layer of photos included in the example called "picture layer," and click the trash can icon to delete it.
2. Create a layer by clicking the Insert Layer button, and name this new layer My Photos.

Make sure that this new layer is the bottom layer.

3. Select the first blank keyframe in the My Photos layer, select File > Import > Import to Stage, and locate your photo sequence.
4. Select the first image in the series, click Open (Windows) or Import (Macintosh), and click Import.
5. Flash recognizes that your image is part of a series and asks you to import all files in the series. Click Yes.

Add finishing touches

Flash places each image on separate keyframes. If you have more than four images, all of the other layers must have an equal number of frames. Your images appear in the Library panel.

1 Delete the old images that were included in this document from the library.

2 Change the title, date, and caption at the top for each image. Replace text as desired.

The template automatically determines how many images are in your document and indicates which photo you are currently using.

Use autoplay mode

The Photo Slideshow template also has a built-in autoplay mode that automatically changes the photo after a set delay. The template is set to a default delay time of 4 seconds, but you can change this setting.

1 Select the controller component on the Stage. It is an instance “mc, controller” component.

2 Open the Component inspector (Window > Component Inspector).

The Parameters tab is selected by default.

3 Select the delay, and change this value to a new delay value, in seconds.

4 Save and publish your document.

Chapter 4: Adobe Version Cue

Adobe Version Cue® CS3 is a file-version manager included with Adobe Creative Suite 3 Design Premium and Standard, Adobe Creative Suite 3 Web Premium and Standard, and Adobe Creative Suite 3 Master Collection. Version Cue enables versioning and asset management in Version Cue-enabled Creative Suite components, including Adobe Acrobat, Adobe Flash, Adobe Illustrator, Adobe InDesign, Adobe Photoshop, Adobe Bridge, and Adobe InCopy.

Working with Adobe Version Cue

About Version Cue

Version Cue is a file-version manager included with Creative Suite 3 Design, Web, and Master Collection editions that consists of two pieces: the Version Cue Server and Version Cue connectivity. The Version Cue Server can be installed locally or on a dedicated computer and hosts Version Cue projects and PDF reviews. Version Cue connectivity, included with all Version Cue-enabled Creative Suite components (Acrobat, Flash, Illustrator, InDesign, InCopy, Photoshop, and Bridge) enables you to connect to Version Cue Servers.

Use Version Cue to track versions of a file as you work and to enable workgroup collaboration such as file sharing, version control, backups, online reviews, and the ability to check files in and out. You can organize Version Cue-managed files into private or shared projects.

Version Cue is integrated with Adobe Bridge: use Bridge as a file browser for Version Cue projects. With Bridge, you can access Version Cue Servers, projects, and files, and view, search for, and compare information about Version Cue-managed assets.

Use Version Cue Server Administration to create and manage user access, projects, and PDF reviews; administer backups; export content; and to specify advanced Version Cue Server information.

For a video on using Version Cue, see www.adobe.com/go/vid0112.

See also

“Version Cue Server Administration” on page 113

“Version Cue PDF reviews” on page 124

What’s new

Initial server configuration When you first start the Version Cue Server (which is turned off by default), the Initial Configuration window in Version Cue Server Administration enables you specify initial server configuration settings.

Improved integration with Adobe Bridge Use the Inspector in Bridge to display and act on context-sensitive information about Version Cue Servers, projects, and assets. Version Cue options in the Content panel in Bridge let you connect to Version Cue Servers, create Version Cue projects, and work with Version Cue-managed assets.

Faster upload/download and more efficient server storage Version Cue transfers and stores only the differences between local files and their counterparts on the Version Cue Server.

Welcome Screen and updated terminology A Welcome screen in Bridge enables you to quickly access Version Cue Servers and projects. Updated terminology makes it easier to work with Version Cue.

New users and groups interface A new interface for managing user access to Version Cue includes the ability to assign permissions based on group membership.

LDAP If your workgroup uses LDAP directories for user account management, you can set up Version Cue to search and add users from these directories. Users can then log in to Version Cue using their LDAP credentials.

SSL Enabling SSL (Secure Sockets Layer, a security protocol), in Version Cue Server Administration allows for secure communication between the Version Cue Server and Version Cue-enabled Creative Suite components.

Version Cue SDK Java developers can use the Version Cue CS3 SDK to create plug-ins that customize workflows or create connections to a DAM (Digital Asset Management) system. The API enables developers to deploy a server-side plug-in to integrate custom solutions into Creative Suite 3 components and Bridge. For more information, see www.adobe.com/go/developer.

See also

"Create projects" on page 95

"Editing and synchronizing offline files" on page 111

"Create and manage users" on page 115

Version Cue basics

Version Cue Server

When you perform the default installation of Creative Suite 3 Design, Web, and Master Collection editions, a *Version Cue Server* is installed on your computer, but is not turned on. Version Cue Servers store Version Cue projects and their related assets. You access the Version Cue Server by using Adobe Bridge or the Adobe dialog box in Version Cue-enabled Creative Suite components. You can start the Version Cue Server on your computer, or, optimally, install and run the Version Cue Server on a dedicated computer accessible to others on your network.

When you first turn on the Version Cue Server, you'll be prompted to specify initial server settings, including a system administrator password, server name and visibility settings, and user account creation settings.

Version Cue Server Administration

Once you've installed and turned on the Version Cue Server, use *Version Cue Server Administration* to set up users, create projects and edit their properties, create and administer PDF reviews, and configure the Version Cue Server.

Version Cue projects

Version Cue uses *projects* to store related files and folders. Projects are stored on Version Cue Servers. Projects store the master copies of files added to the project, as well as file metadata such as version information and comments.

Local project files and server versions

Local project files are created on your hard drive when you open and edit a file from a Version Cue project (Version Cue marks the file as checked out by you). As you work with the local project file, you save changes to it by choosing File > Save. This updates the local file on your hard drive, but not the file on the Version Cue Server.

When you're ready to check in the local project file changes back to the Version Cue Server, you create a *version* by using the Check In command. Versions represent a snapshot of the file at a given time.

The Version Cue Server stores all versions of a file so you can view earlier versions, promote earlier versions to be the current version, or delete unnecessary or obsolete versions.

Version control

Version Cue allows multi-user access to files on the Version Cue Server. If two users try to edit a file on the Version Cue Server, Version Cue institutes *version control* by notifying the second user that the file is checked out. Version Cue then lets you decide how to proceed.

See also

"Version Cue Server Administration" on page 113

"Working with Version Cue projects" on page 94

"Version Cue Server Administration" on page 113

"Version Cue versions" on page 108

"Edit files checked out by another user" on page 103

Version Cue workflow

Before you begin using Version Cue features, you'll need to install and configure the Version Cue Server, create a project, and assign users to it.

1. Install and configure the Version Cue Server

When you install Creative Suite 3 Design, Web, and Master Collection editions, a Version Cue Server is installed on your computer, but is not turned on. You can turn on the server to enable simple file sharing; however, if you want to share Version Cue-managed assets with a workgroup, you should install it on a dedicated computer accessible to others on your network.

When you start the server for the first time, you'll be prompted to specify initial server settings, including a system administrator password, server name and visibility settings, and default user access rights. See "About Version Cue Server Administration" on page 113

Configure the server further by specifying settings in Version Cue Server preferences and in Version Cue Server Administration. See "Version Cue Server Administration" on page 113 and "Advanced Version Cue Server Administration tasks" on page 122.

2. Create a project and assign users

After you've set up and configured the Version Cue Server, you can create projects and assign users to them. By default, projects you create in Version Cue are private. You change a project's shared status at any time, and restrict access to the project, by specifying that users log in when they access the project.

Create projects by using Bridge, the Adobe dialog box, or Version Cue Server Administration. To specify advanced project properties, such as requiring user login and assigning user access permissions, you must use Version Cue Server Administration. See "Create projects" on page 95 and "Create and manage projects in Version Cue Server Administration" on page 118.

3. Add files to a project

Once you've created a project, add files to the project so users can check them out, make changes, and check them back in. You can add multiple Adobe or non-Adobe files by using Bridge, or add files one at a time from within a Version Cue-enabled Creative Suite component by using the Adobe dialog box. See "Add files and folders to a project" on page 98

Accessing Version Cue features

Access to Version Cue features, by way of the Adobe dialog box or Bridge, varies depending on whether or not you use Version Cue-enabled software and whether or not you use one of the Adobe Creative Suite products (for example, Adobe Creative Suite Design Premium).

For instance, you have access to the full feature set, either through the Adobe dialog box or Bridge, when you use Photoshop as part of a suite product. By contrast, if you use Photoshop as standalone software, you must be granted access to a shared project in order to use the full Version Cue feature set. In Dreamweaver, Contribute, and Fireworks, you have access to Version Cue features only through Bridge. The following table explains the scenarios in which you have access to Version Cue features, and how you access those features.

Software component	Access via the Adobe dialog box	Access via Bridge
Acrobat, Bridge, Illustrator, InCopy, InDesign, Photoshop, Flash	When used as part of a Creative Suite product: Yes When used as standalone software: Only if granted access to a shared project	When used as part of a Creative Suite product: Yes When used as standalone software: Only if granted access to a shared project
Dreamweaver, Contribute, Fireworks,	No	When used as part of a Creative Suite product: Yes When used as standalone software: Only if granted access to a shared project

Adobe recommends managing non-Adobe files with Bridge. However, if you're collaborating with other users who don't have access to Bridge, you can use the Version Cue WebDAV Server URL to access projects on a Version Cue Server.

IT administrators can use the Adobe Version Cue Access Utility, available for download from the Adobe website, to access and extract current versions of files stored in a Version Cue project. For more information, visit the Adobe website.

Using Creative Suite 2 components and Acrobat 8 with Version Cue CS3

You can use Adobe Creative Suite 2 components and Acrobat 8 with the Version Cue CS3 Server; however, there are some differences to keep in mind.

- If you're using Acrobat 8 or an Adobe Creative Suite 2 component to access Version Cue CS3-managed files, the files must be part of a Version Cue CS2-compatible project. You can specify that a Version Cue CS3 project is Version Cue CS2-compatible when you create it. (You cannot specify that a project is Version Cue CS2-compatible after you create it.)

***Note:** Projects migrated from Version Cue CS2 to Version Cue CS3 remain compatible with Acrobat 8 and Adobe Creative Suite 2 components.*

- Acrobat 8 and Adobe Creative Suite 2 components can't connect to a Version Cue CS3 Server if you enable SSL in Version Cue Server Administration.
- Adobe Creative Suite 2 components can't work with Version Cue CS3 servers that are installed on the same computer. Adobe Creative Suite 2 components can, however, connect to Version Cue CS3 Servers that reside on the network.
- The Version Cue CS2 Workspace and the Version Cue CS3 Server can be installed and function on the same computer simultaneously (and must be installed on the same computer if you want to migrate projects from Version Cue CS2 to Version Cue CS3).
- If a Version Cue CS2 Workspace and a Version Cue CS3 Server are installed on the same computer, Adobe Creative Suite 2 components work only with the Version Cue CS2 Workspace, because they can communicate only with the port that the Version Cue CS2 Workspace uses.
- Version Cue CS3 doesn't support alternates; however, Adobe Creative Suite 2 components can work with alternates in Version Cue CS2-compatible projects on a Version Cue CS3 Server. Adobe Creative Suite 3 components cannot access alternates in a Version Cue CS2-compatible project on a Version Cue CS3 Server.

For help using Acrobat 8 with Version Cue CS2, see "Using Version Cue" in Acrobat 8 Help. For help using Adobe Creative Suite 2 components with Version Cue CS2, see Version Cue CS2 Help.

See also

"Create and manage projects in Version Cue Server Administration" on page 118

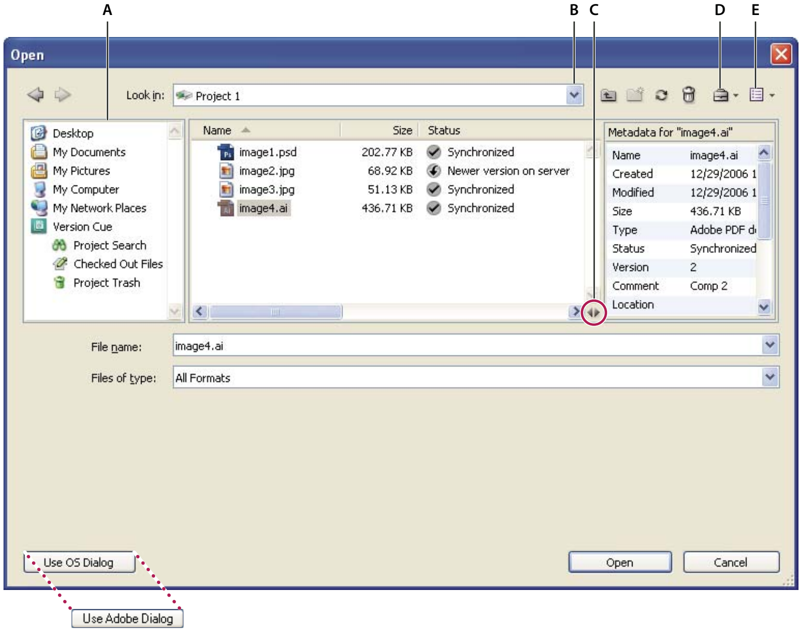
Use the Adobe dialog box

In Version Cue-enabled Creative Suite components, you can use the Adobe dialog box when you choose the Open, Import, Export, Place, Save, or Save As commands. The Adobe dialog box gives you access to Version Cue commands and controls, and displays thumbnails and other information that makes it easy to identify files.

To use the Adobe dialog box, click Use Adobe Dialog in the Open, Import, Export, Place, Save, or Save As dialog boxes.

If the Use Adobe Dialog button doesn't appear in the Open, Import, Export, Place, Save, or Save As dialog box, make sure that you have enabled Version Cue file management in Bridge or in the Creative Suite component you're using.

Use the View menu options to customize the display. You can change back to the OS dialog box at any time by clicking Use OS Dialog.



Look in: Project 1

Desktop
My Documents
My Pictures
My Computer
My Network Places
Version Cue
Project Search
Checked Out Files
Project Trash

Name	Size	Status
image1.psd	202.77 KB	Synchronized
image2.jpg	68.92 KB	Newer version on server
image3.jpg	51.13 KB	Synchronized
image4.ai	436.71 KB	Synchronized

Metadata for "image4.ai"

Name	image4.ai
Created	12/29/2006 1
Modified	12/29/2006 1
Size	436.71 KB
Type	Adobe PDF d
Status	Synchronized
Version	2
Comment	Comp 2
Location	

File name: image4.ai

Files of type: All Formats

The Adobe dialog box
***A.** Favorites panel* ***B.** Look In menu* ***C.** Toggle metadata* ***D.** Tools menu* ***E.** View menu*

Version Cue Server and project icons

Bridge and the Adobe dialog box display status icons for Version Cue Servers and projects to let you know whether they're online (available), offline, local, or remote.

Shared Project Indicates a project that's available and shared with other users.

Private Project Indicates a project that's available and not shared with other users.

Offline Project Indicates a project that's not available.

VC2 Compatible Project Indicates a project that is compatible with Adobe Creative Suite 2 components and Adobe Acrobat 8.

Offline Server Indicates an offline Version Cue Server.

My Server Indicates a Version Cue Server that is local to your computer.

Network Server Indicates a remote Version Cue Server that's available.

Enable Version Cue file management

Version Cue file management, which provides access to Version Cue projects, is enabled by default in Bridge CS3 and all Version Cue-enabled Creative Suite components, except for Acrobat 8. (You must always enable or disable Version Cue file management manually in Acrobat 8.)

If you disable Version Cue file management in one Creative Suite component, you disable it in all other Version Cue-enabled Creative Suite components, except Acrobat and Bridge. If you disable Version Cue file management in Bridge, you disable it in all Version Cue-enabled Creative Suite components, except Acrobat.

Disabling Version Cue file management means that you disable access to all Version Cue projects on all Version Cue Servers.

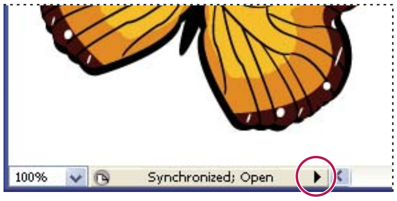
- In the Startup Scripts preferences in Bridge, select Version Cue, and click OK.
- In File Handling & Clipboard preferences in Illustrator, select Enable Version Cue, and click OK.
- In File Handling preferences in InDesign, select Enable Version Cue, and click OK.
- In File Handling preferences in Photoshop, select Enable Version Cue, and click OK.
- In General preferences in Flash, select Enable Version Cue, and click OK.
- In Documents preferences in Acrobat, select Enable Version Cue File Version Manager, and click OK.
- In File Handling preferences in InCopy, select Enable Version Cue, and click OK.

View Version Cue information

Note: *You can perform this task only if you have access to the full Version Cue feature set. See "Accessing Version Cue features" on page 85.*

You can view information about Version Cue Servers, projects, and assets in the Adobe dialog box or by using the Inspector in Bridge. For help on viewing information in Bridge, see "Inspect Version Cue files" in Bridge Help.

If you've already opened a Version Cue-managed file in a Version Cue-enabled Creative Suite component, you can view information about it in the status bar at the lower-left of the document window (in Acrobat, this information appears in the lower-left of the navigation pane).



Status bar

1 In Acrobat, Flash, Illustrator, InCopy, InDesign, or Photoshop, choose File > Open.

2 Click Use Adobe Dialog. (If you see Use OS Dialog instead, you are already using the Adobe dialog box.)

3 Click Version Cue in the Favorites panel.

4 To change the display of Version Cue Servers, projects, or files in the dialog box, choose a display option from View menu ☰ .

Note: *If a Version Cue Server for which you want to view information is outside your subnet, use the Connect To Server command from the Tools menu (or Connect To from the Project Tools menu in Acrobat) to access it.*

5 To display information about a Version Cue Server, project, or file, do one of the following:

- Click the toggle ◀▶ to display the Properties panel and view the properties of a file.
- Place the pointer over the item. Information appears in a tool tip.
- Select the file and choose Versions from the Tools or Project Tools (Acrobat) menu to display information about a file's versions.

See also

“File statuses” on page 101

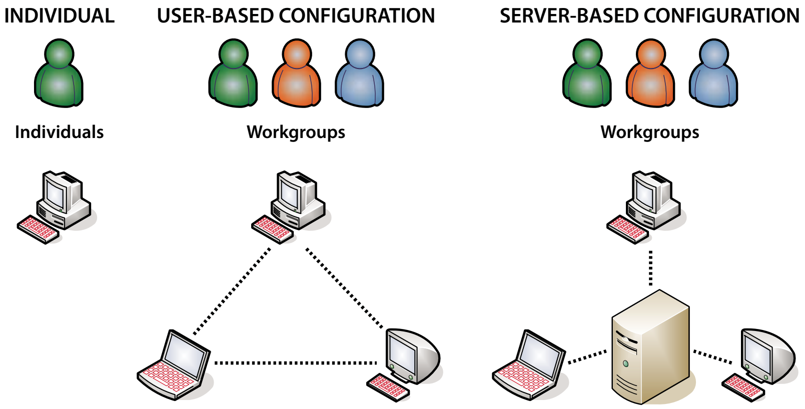
Working with the Version Cue Server

About Version Cue Server installation

Version Cue Servers store Version Cue projects and their related assets. When you perform a default installation of Creative Suite 3 Design, Web, or Master Collection editions, Version Cue installs the Version Cue Server on your computer, but does not turn it on. When the Version Cue Server is installed on your computer, the server is available only if your computer is turned on and networked to the other users in your group. This scenario is adequate for personal use or for file sharing between individuals.

Alternatively, you can install the Version Cue Server on a dedicated computer accessible to others on your network, so that Version Cue-managed assets are always available to a workgroup.

For a video on using Version Cue in a workgroup, see www.adobe.com/go/vid0113.



Local or server-based file sharing: Version Cue can be set up to operate in a user- or server-based environment. In a user-based implementation, files and projects are shared from an individual’s own hard drive. In a server-based environment, the Version Cue Server resides on a separate, dedicated computer.

The Version Cue Server is installed in the Program Files/Common Files/Adobe/Adobe Version Cue CS3/Server folder (Windows) or in the Library/Application Support/Adobe/Adobe Version Cue CS3/Server folder (Mac OS). You cannot change this location.

To install the Version Cue Server on a dedicated computer, run the Creative Suite 3 Design, Web, or Master Collection edition installer on the dedicated computer, following the on-screen prompts to install only the Version Cue Server.

***Note:** Consult the End-User License Agreement (EULA) for your copy of Adobe Creative Suite before installing the Version Cue Server on a dedicated computer.*

Turn on and configure the Version Cue Server

To use a Version Cue Server, you’ll need to turn it on and configure initial settings. Once you’ve configured initial settings, you can configure additional Version Cue Server settings in the Version Cue preferences and specify advanced server settings (such as enabling SSL) in Version Cue Server Administration.

For a video on setting up the Version Cue server, see www.adobe.com/go/vid0114.

1 Do either of the following:

- Click Start My Server in the Adobe dialog box or in Adobe Bridge.
- Open the Control Panel and double-click Adobe Version Cue CS3 (Windows) or click Adobe Version Cue CS3 in System Preferences (Mac OS), and then click Start.

Version Cue starts Version Cue Server Administration and displays the Initial Configuration window.

2 In the Initial Configuration window, specify a system administrator password in the Password box.

Note: *Be sure to note the password you specify. If you forget the system administrator password, you'll need to reinstall the Version Cue Server.*

3 Specify a name for the server in the Server Name box.

4 Choose an option from the Server Visibility menu:

- To prevent other users in your network from seeing the server, choose Private. Private Version Cue Servers can be accessed only from your local computer.
- To make the server visible to other users in your network, choose Visible To Others. (You must configure the server to be visible to grant others access to projects on the server.)

Note: *If Version Cue is installed on a Windows computer that uses a firewall and you want to share the server with others, make sure that TCP ports 3703 and 5353 are left open. If you've enabled SSL for the Version Cue Server, also leave port 3704 open. If Version Cue CS2 is installed on the same computer, also leave port 50900 open (and 50901 if you've enabled SSL). For instructions, see Windows Help.*

5 Choose an option from the User Accounts menu: I

- To enable users to access the server without an existing user account, choose Automatic User Creation. If you select this option, Version Cue creates a new user account without a password when a new user accesses the server.

If you enable Automatic User Account Creation and then subsequently enable LDAP support, LDAP users are automatically imported when they access the server with their LDAP account name. Users imported in this fashion are added to the Everyone group, given a user access level of None, and are not able to log into Version Cue Server Administration. Use this technique to automatically assign LDAP users default access rights to projects on a Version Cue server without having to explicitly import users.

- To specify that only named users, defined in Version Cue Server Administration, can access the server, choose Manual User Creation.

6 Click Save & Continue to log in to Version Cue Server Administration and specify advanced server settings.

See also

"Version Cue Server Administration" on page 113

"Use the Adobe dialog box" on page 86

"Create and manage users" on page 115

Set Version Cue Server preferences

You can configure many Version Cue Server settings in Version Cue preferences, such as the amount of RAM available to Version Cue and the location of the Data folder. To configure advanced settings, such as enabling SSL, changing the name of the Version Cue Server, specifying server log options, resetting user locks, or backing up the server, you must use Version Cue Server Administration.

See also

"Advanced Version Cue Server Administration tasks" on page 122

Access Version Cue Server preferences

1 Do one of the following to access Version Cue preferences:

- In Windows, double-click the Version Cue icon in the system tray at the lower-right of the screen.
- In Mac OS, click the Version Cue icon in the menu bar at the top of the screen, and choose Version Cue CS3 Preferences from the menu.
- Open the Control Panel and double-click Adobe Version Cue CS3 (Windows) or click Adobe Version Cue CS3 in System Preferences (Mac OS).

2 Click the Settings tab in the Adobe Version Cue CS3 dialog box.

Make the Version Cue Server visible

1 To grant others access to shared Version Cue projects on the server, choose This Server Is Visible To Others from the Server Visibility menu. To hide the Version Cue Server from other users, choose This Server Is Private.

***Note:** If Version Cue is installed on a Windows computer that uses a firewall and you want to share the server with others, make sure that TCP ports 3703 and 5353 are left open. If you've enabled SSL for the Version Cue Server, also leave port 3704 open. If Version Cue CS2 is installed on the same computer, also leave port 50900 open (and 50901 if you've enabled SSL). For instructions, see Windows Help.*

2 Click Apply.

Specify a workgroup size

1 From the Workgroup Size menu, choose the number of people who use the Version Cue Server on a typical day. This setting controls how the Version Cue Server handles the potential load.

2 Click Apply.

Specify RAM

The default amount of allocated RAM (128 MB) is sufficient for workgroups of fewer than 10 people and projects with fewer than 1000 assets. Allocate at least 256 MB of RAM for larger workgroups and projects with up to 1000 assets. Allocate at least 512 MB of RAM if you work with more than 1000 assets per project or more than 50 projects, regardless of workgroup size.

1 In the Memory Usage box, enter the amount of RAM that you want to make available to Version Cue (the default is 128 MB).

2 Click Apply.

Keep the Version Cue icon visible

1 Select Show Version Cue CS3 Tray Icon (Windows) or Show Version Cue CS3 Status in Menu Bar (Mac OS) to keep the Version Cue icon visible.

2 Click Apply.

Turn Version Cue on when the computer starts

1 Select Turn Version Cue CS3 On When The Computer Starts.

2 Click Apply.

Change the location of the Data folder

The Data folder contains files that maintain the integrity of Version Cue projects, file versions, and metadata. You can change the location of the Data folder; however, you cannot move it to a network volume. If you move the Data folder to an external disk in Mac OS, make sure to deselect Ignore File Permissions in the disk's Get Info dialog box.

Important: *Shut down the Version Cue Server before you change the folder location. Do not attempt to move this folder manually or edit any of the files in the Version Cue Data folder.*

1 Do one of the following to access Version Cue preferences:

- In Windows, double-click the Version Cue icon in the system tray at the lower-right of the screen.
- In Mac OS, click the Version Cue icon in the menu bar at the top of the screen, and choose Version Cue CS3 Preferences from the menu.
- Open the Control Panel and double-click Adobe Version Cue CS3 (Windows) or click Adobe Version Cue CS3 in System Preferences (Mac OS).

2 Click the Locations tab in the Adobe Version Cue CS3 dialog box.

3 Click the Choose button next to the current Data folder location, and select a new location for the folder. You must choose a location on the computer (including external disks) where the Version Cue Server is installed.

4 Click OK.

5 Click Apply. If prompted, click Yes (Windows) or Restart (Mac OS) to restart the Version Cue Server.

See also

"Shut down or restart the Version Cue Server" on page 94

Connect to remote servers

Note: *You can perform this task only if you have access to the full Version Cue feature set. See "Accessing Version Cue features" on page 85.*

When you need to work on Version Cue projects that are located on a different subnet, you can use the IP address of the computer to access the remote Version Cue Server, as long as it is configured to be visible to other users. Version Cue Servers within your subnet that are configured to be visible are visible automatically.

1 Do one of the following:

- In Acrobat, Flash, Illustrator, InCopy, InDesign, or Photoshop, choose File > Open. If you're using the OS Dialog box, click Use Adobe Dialog. Click Version Cue in the Favorites panel, and choose Connect To Server from the Tools menu or Connect To from the Project Tools (Acrobat) menu .
- In Bridge, choose Tools > Version Cue > Connect To Server.

2 In the Connect To Server dialog box, type the IP or DNS address and port of the Version Cue Server, for example, http://153.32.235.230. If you're connecting to a server that is on the same system as a Version Cue CS2 workspace, append the port number 50900 to the end of the Version Cue URL, for example, http://153.32.235.230:50900.

Display the Version Cue Server Administration login page to identify the Version Cue URLs that remote users and WebDAV applications need to access the server. Alternatively, view the URLs in the Inspector in Bridge.

3 Click OK.

A shortcut to the remote server is automatically included in your list of available Version Cue Servers.

Connect to a Version Cue Server using WebDAV

Adobe recommends managing non-Adobe files with Adobe Bridge. However, if you're collaborating with other users who don't have access to Adobe Bridge, you can use the Version Cue WebDAV Server URL to access projects on a Version Cue Server.

You can access a Version Cue Server by using a WebDAV-enabled application, such as a Microsoft Office application. In Windows, specify a project on a Version Cue Server as a network place by specifying the project's WebDAV URL. In Mac OS, specify the project's WebDAV URL by using the Connect To Server Command from the Finder. Before attempting to connect, refer to your application's documentation on using its WebDAV features.

❖ Enter the Version Cue WebDAV URL, the port number (3703, or 50900 if you're connecting to a server that is running on the same system as a Version Cue CS2 workspace), "webdav," and the project name. For example: http://153.32.235.230:3703/webdav/*project_name*

Migrate projects to the Version Cue 3.0 Server

If you currently use Version Cue CS2, you need to migrate your projects to Version Cue CS3. When you migrate Version Cue CS2 projects to Version Cue CS3, users assigned to those projects are also migrated.

You cannot migrate Version Cue CS2 projects to Version Cue CS3 on Intel-based Macintosh computers.

Before migrating projects, ask all users to synchronize their assets so project data is up to date.

1 Locate the folder "com.adobe.versioncue.migration_2.0.0" on the computer on which Version Cue CS3 is installed and copy it to the Version Cue CS2 Plugins folder.

2 Restart Version Cue CS2.

3 Log in to Version Cue CS3 Server Administration.

4 Click the Advanced tab, and then click Import Version Cue CS2 Data.

5 Enter a Version Cue CS2 administrator login and password, and click Log In.

6 Select the project you want to migrate, and click Migrate.

***Note:** If the Version Cue CS2 project has the same name as a project that exists on the Version Cue CS3 Server, Version Cue will append a number to the end of the Version Cue CS2 project name (for example, Test Project (2)). If a Version Cue CS2 user has the same user name as an existing user on the Version Cue CS3 Server, Version Cue will use the existing Version Cue CS3 user account.*

7 When Version Cue Server Administration displays the confirmation page, click End.

8 Stop the Version Cue CS2 workspace.

9 Uninstall Version Cue CS2.

10 Restart the Version Cue CS3 Server. This resets the port to allow access from both Adobe Creative Suite 2 and Adobe Creative Suite 3 components.

See also

"Log in to Version Cue Server Administration" on page 114

Shut down or restart the Version Cue Server

When you shut down the Version Cue Server, you disable access to the Version Cue projects hosted on that server.

Each time you restart the Version Cue Server, it performs an integrity check and makes repairs, if necessary. To ensure best performance, restart the Version Cue Server weekly so that it can perform the integrity check and make repairs.

1 Do one of the following to access Version Cue preferences:

- In Windows, double-click the Version Cue icon in the system tray at the lower-right of the screen.
- In Mac OS, click the Version Cue icon in the menu bar at the top of the screen, and choose Version Cue CS3 Preferences.
- Open the Control Panel and double-click Adobe Version Cue CS3 (Windows) or click Adobe Version Cue CS3 in System Preferences (Mac OS).

2 Click the Settings tab in the Adobe Version Cue CS3 dialog box.

- To shut down the Version Cue Server, click Stop. When prompted, click Yes (Windows) or Shut Down (Mac OS).
- To restart the Version Cue Server, click Stop, and then click Start.
- To automatically turn on Version Cue when the computer starts, select Turn Version Cue CS3 On When The Computer Starts.

3 Click OK (Windows) or Apply Now (Mac OS).

You can also restart the Version Cue Server by clicking Restart Server in the Advanced tab of Version Cue Server Administration.

See also

"Advanced Version Cue Server Administration tasks" on page 122

Working with Version Cue projects

About Version Cue projects

Version Cue projects are stored on Version Cue Servers. Projects store the master copies of files added to the project, as well as file versions and other file data, such as comments and version dates. When the Version Cue Server is specified to be visible and projects are shared, multiple users can access projects, which can contain both Adobe and non-Adobe files.

When you first open a Version Cue project, Version Cue creates a folder named "Version Cue" in your My Documents (Windows) or Documents (Mac OS) folder, and adds a project folder to the Version Cue folder. Version Cue also creates a shortcut to the project that appears in Bridge and in the Adobe dialog box after you click the Version Cue favorite icon.

You can create and administer projects only if you've been assigned appropriate permissions in Version Cue Server Administration.

***Note:** If you use an Adobe Creative Suite 2 component or Acrobat 8, you won't be able to see Version Cue CS3 projects in the Adobe dialog box or in Bridge unless the project is specified to be backward compatible with Adobe Creative Suite 2 and Acrobat 8. In addition, Creative Suite 2 components and Acrobat 8 can't connect to a Version Cue CS3 Server that uses SSL.*

See also

"About local project files" on page 100

"About versions" on page 108

"Create and manage users" on page 115

Create projects

***Note:** You can perform this task only if you have access to the full Version Cue feature set. See "Accessing Version Cue features" on page 85.*

You can create projects by using Bridge, any Version Cue-enabled Creative Suite component, or Version Cue Server Administration, which provides options for specifying advanced project properties.

You must have Project Administration permissions to be able to create projects in Version Cue.

See also

"Create and manage users" on page 115

Create a project

1 In Acrobat, Flash, Illustrator, InCopy, InDesign, or Photoshop, choose File > Open. Click Use Adobe Dialog if you're using the OS dialog box.

2 Click Version Cue in the Favorites panel.

3 Choose New Project from the Tools or Project Tools (Acrobat) menu .

4 Choose a Version Cue Server to host the project from the Location menu.

5 Enter a name for the project in the Project Name box and a description in the Project Info box.

6 To make this project and its files available to others, select Share This Project With Others.

7 To create a project that Creative Suite 2 or Acrobat 8 users can access, select Maximize Compatibility With CS2 Applications And Acrobat 8.

8 Click OK.

Create a project in Bridge

1 Choose Tools > Version Cue > New Project.

2 In the New Project dialog box, choose a Version Cue Server to host the project from the Location menu.

3 Enter a name for the project in the Project Name box and a description in the Project Info box.

4 To make this project and its files available to others, select Share This Project With Others.

5 To create a project that Creative Suite 2 or Acrobat 8 users can access, select Maximize Compatibility With CS2 Applications And Acrobat 8.

6 Click OK.

Open a project

***Note:** You can perform this task only if you have access to the full Version Cue feature set. See "Accessing Version Cue features" on page 85.*

You can open projects that are stored on a local Version Cue Server or open shared projects on a remote server that is configured to be visible to others.

See also

"Use the Adobe dialog box" on page 86

Open a project

1 In Acrobat, Flash, Illustrator, InCopy, InDesign, or Photoshop, choose File > Open. Click Use Adobe Dialog if you're using the OS dialog box.

2 Click Version Cue in the Favorites panel.

3 Do one of the following:

- Double-click your local server to view its projects.
- Double-click Recent Projects to open a recently opened project.
- Double-click Browse Servers to locate the Version Cue Server that hosts the project. When you locate the server, double-click it to view its projects.

***Note:** If the server that hosts the project is outside your network, choose Connect To Server from the Tools menu or Connect To from the Project Tools menu (Acrobat), enter the IP or DNS address of the server, and click Connect.*

4 Double-click the project to open it.

Open a project in Bridge

1 Click Version Cue in the Favorites panel.

2 Do one of the following:

- Double-click your local server to view its projects.
- Double-click Recent Projects to open a recently opened project.
- Double-click Browse Servers to locate the Version Cue Server that hosts the project. When you locate the server, double-click it to view its projects.

***Note:** If the server that hosts the project is outside your subnet, choose Tools > Version Cue > Connect To Server, enter the IP or DNS address of the server, and click Connect.*

3 Double-click the project to open it.

Edit project properties

***Note:** You can perform this task only if you have access to the full Version Cue feature set. See "Accessing Version Cue features" on page 85.*

You can edit most of the properties of a project, including the project name, description, and shared status; the location of local project files; and the location of project backups in Bridge or in the Adobe dialog box. However, if you want to enable lock protection, edit or assign users, or require users to log in to the project, use Version Cue Server Administration instead.

See also

"Use the Adobe dialog box" on page 86

"Open a project" on page 96

"About local project files" on page 100

"Create and manage projects in Version Cue Server Administration" on page 118

Access project properties

- In Acrobat, Flash, Illustrator, InCopy, InDesign, or Photoshop, select the project in the Adobe dialog box, and then choose Edit Properties from the Tools or Project Tools (Acrobat) menu.
- In Bridge, select the project, choose View > Inspector Panel, and click Edit Properties in the Tasks area of the Inspector panel.
- In Bridge, select the project and then choose Tools > Version Cue > Edit Properties.

***Note:** To edit advanced project properties in Version Cue Server Administration, click Server Administration in the Edit Properties dialog box.*

Change the project name or description

1 In the Edit Properties dialog box, enter a name in the Project Name box. The new name will not be reflected in your (or your workgroup's) local project folder until you disconnect from and reconnect to the project. To change the description of the project, enter text in the Project Info box.

2 Click Save.

Change the location of local project files

1 In the Edit Properties dialog box, expand Local Project Files to view the location of local project files on your computer.

2 Click Change Location and choose the new location for local project files on your computer.

***Note:** Do not move the project folder manually in the file system to change the location of local project files.*

3 Click Save.

Change a project's shared status

***Note:** You can perform this task only if you have access to the full Version Cue feature set. See "Accessing Version Cue features" on page 85.*

By default, Version Cue CS3 projects are private; however, you can change a project's shared status at any time.

You can use Version Cue Server Administration to require login for shared projects, thus restricting access to specific users.

See also

"Use the Adobe dialog box" on page 86

"Open a project" on page 96

"Create and manage projects in Version Cue Server Administration" on page 118

Share or unshare a project

1 In Acrobat, Flash, Illustrator, InCopy, InDesign, or Photoshop, select the project in the Adobe dialog box, and then choose Edit Properties from the Tools or Project Tools (Acrobat) menu.

2 Select or deselect Share This Project With Others, and click Save.

Share or unshare a project from Bridge

1 Click Version Cue in the Favorites panel.

2 Select the project, and do one of the following:

- Choose View > Inspector Panel, and click Edit Properties in the Tasks area of the Inspector panel.
- Choose Tools > Version Cue > Edit Properties.

3 Select or deselect Share This Project With Others, and click Save.

Change the location of project backups

Note: *You can perform this task only if you have access to the full Version Cue feature set. See "Accessing Version Cue features" on page 85.*

You must shut down the Version Cue Server before you change the backup folder location. Do not move this folder manually.

1 Shut down the Version Cue Server (see "Version Cue Server Administration" on page 113.

2 Do one of the following to access Version Cue preferences:

- In Windows, double-click the Version Cue icon in the system tray at the lower-right of the screen.
- In Mac OS, click the Version Cue icon in the menu bar at the top of the screen, and choose Version Cue CS3 Preferences from the menu.
- Open the Control Panel and double-click Adobe Version Cue CS3 (Windows) or click Adobe Version Cue CS3 in System Preferences (Mac OS).

3 Click the Locations tab in the Adobe Version Cue CS3 dialog box.

4 Click the Choose button next to the Backup Folder location, and select a new location for the folder. You must choose a location on the computer on which the Version Cue Server is installed.

5 Click OK.

6 Click OK (Windows) or Apply Now (Mac OS). If prompted, click Yes (Windows) or Restart (Mac OS) to restart the Version Cue Server.

Add files and folders to a project

Note: *You can perform this task only if you have access to the full Version Cue feature set. See "Accessing Version Cue features" on page 85.*

To save versions of a file, share the file with your workgroup, and take advantage of Version Cue file management, you must add the file to a Version Cue project. You can add both Adobe and non-Adobe files to Version Cue projects. Add files one at a time from within a Version Cue-enabled Adobe Creative Suite component by using the Adobe dialog box, or add groups of files by using the Add Files command in Bridge.

See also

"Use the Adobe dialog box" on page 86

"Open a project" on page 96

Add a file to a project

1 Open the file in Acrobat, Flash, Illustrator, InCopy, InDesign, or Photoshop.

2 Choose File > Save As. Click Use Adobe Dialog if you're using the OS dialog box.

3 Click Version Cue in the Favorites panel, open the project to which you want to add the file, and click Save As.

4 Enter a version comment in the Check In or Save A Version (Acrobat) dialog box and click OK.

Add a file or folder to a project in Bridge

1 Click Version Cue in the Favorites panel and open the project to which you want to add files.

2 Do one of the following:

- Drag files or folders from Explorer (Windows) or the Finder (Mac OS) to the project in Bridge. (You cannot drag empty folders to a Version Cue project in Bridge.)
- Choose Tools > Version Cue > Add Files. In the Open dialog box, select one or more files and click Open.

3 Enter a version comment in the Check In dialog box and click OK.

Delete projects

***Note:** You can perform this task only if you have access to the full Version Cue feature set. See "Accessing Version Cue features" on page 85.*

You can delete a Version Cue project in Bridge or any Version Cue-enabled Adobe Creative Suite component, or by using Version Cue Server Administration. You can't delete a project if any user has files that are marked as Checked Out or In Use (Acrobat).

***Note:** Remove file locks to remove the Checked Out or In Use (Acrobat) status of files designated as such. See "Advanced Version Cue Server Administration tasks" on page 122.*

Deleting a project permanently erases all of its files (including versions) and folders from the Version Cue Server, and erases shortcuts to the project and the local project files on your computer. (The local project files created on other users' computers are not deleted until they disconnect from the deleted project.)

See also

"Use the Adobe dialog box" on page 86

"Open a project" on page 96

"Create and manage projects in Version Cue Server Administration" on page 118

Delete a project

❖ In Acrobat, Flash, Illustrator, InCopy, InDesign, or Photoshop, select the project you want to delete in the Adobe dialog box, and then click the Delete icon 🗑.

Delete a project in Bridge

1 Click Version Cue in the Favorites panel.

2 Select the project you want to delete, and then click the Delete Item icon 🗑.

Disconnect from projects

Note: *You can perform this task only if you have access to the full Version Cue feature set. See "Accessing Version Cue features" on page 85.*

Disconnecting from a project erases the local project files on your computer, while leaving the master copies on the Version Cue Server intact. Disconnecting also removes shortcuts to the project from Bridge and the Adobe dialog box. You may want to disconnect to free up more space on your hard drive (new local project files are created the next time you open, download, edit, or synchronize a file). You may also disconnect from a project to erase your local project files from a project deleted by someone else in your workgroup.

If an administrator deletes a project in which you have local project files with Checked Out or In Use (Acrobat) status, you must manually delete the local project files folder from your hard drive. You can then disconnect from the project.

See also

"Use the Adobe dialog box" on page 86

"Open a project" on page 96

Disconnect from a project

1 In Acrobat, Flash, Illustrator, InCopy, InDesign, or Photoshop, click Version Cue in the Favorites panel of the Adobe dialog box.

2 Right-click (Windows) or Control-click (Mac OS) the project from which you want to disconnect, and then choose Disconnect.

Disconnect from a project in Bridge

1 Click Version Cue in the Favorites panel.

2 Right-click (Windows) or Control-click (Mac OS) the project from which you want to disconnect, and then choose Disconnect.

Working with files in Version Cue

About local project files

When you work in files from a Version Cue project, you're editing a *local copy* of the file in the project folder on your hard drive, not the master file on the Version Cue Server, which remains protected and untouched. Local project files also allow you to work on a file simultaneously with others.

Important: *To relocate local project files on your hard drive, use the Change Location feature (don't move the project folder manually in the file system). For instructions, see "Edit project properties" on page 96.*

As you work, use the Save command to save changes periodically and update your local project file. A new version is added to the master file on the Version Cue Server when you choose the Check In or Save A Version (Acrobat) command, or when you synchronize your files with the Version Cue Server.

See also

"Accessing Version Cue features" on page 85

"About Version Cue projects" on page 94

"About versions" on page 108

"Synchronize files" on page 112

File statuses

Files that are managed by Version Cue are marked with a status icon that describes the state of the file on the Version Cue Server. You can view a file's status while browsing the files in a Version Cue project, in Bridge, and also in the document window's status area after opening a file in a Version Cue-enabled Creative Suite component (in Acrobat, the status is displayed in the lower-left corner of the navigation pane).

If you don't see the Version Cue status in the document window's status area, click the status bar and choose Show > Version Cue Status.

A file can have more than one status at the same time.

Open The file is open on your computer. The Open status is indicated only for files on your computer.

Checked Out By Me You are editing the file. Version Cue assigns this status when you make an edit that changes the file's content. You can mark a file as checked out before you edit it to alert other users that you intend to make changes to the content.

Checked Out By <user name> Another user is editing the file and has not yet saved a new version.

To quickly view all files checked out in a particular project, open the project and click Checked Out Files beneath the Version Cue entry in the Favorites panel of the Adobe dialog box.

Synchronized The latest known version of the file is available for editing and you have a local copy of it on your computer. Version Cue assigns this status when you check in a version of the file you're editing, or when you synchronize a project.

Conflicting There is a version conflict, or both you and another user are editing the file.

New File The file in the local project folder is the only copy known to Version Cue and has not been synchronized with the Version Cue Server. This status may be applied, for instance, if a file is saved in an existing project for the first time while the Version Cue Server is offline. You can edit the file, but it's important to check in or synchronize the file after you save your changes.

Newer Version On Server A A local project file exists, but there is a newer version of the file on the Version Cue Server. This status indicates that it will take a few moments to download an up-to-date local project file before you can edit the file.

Server Offline There is a local project file, but the Version Cue Server is offline, or you are offline and not able to access the server. There is no way of checking whether the local project file is synchronized with the latest version on the Version Cue Server. You can edit an offline copy and save these changes; however, you must check in a version or synchronize the file when the Version Cue Server comes back online.

Deleted The file or folder has been deleted from the project, but not yet permanently erased. (You can restore a deleted file or folder.)

See also

“Accessing Version Cue features” on page 85

“Editing and synchronizing offline files” on page 111

“Delete files or folders from a project” on page 107

“Edit files checked out by another user” on page 103

Open a file in a project

***Note:** You can perform this task only if you have access to the full Version Cue feature set. See “Accessing Version Cue features” on page 85.*

You can open files only from projects that are stored on a local Version Cue Server or from shared projects on a remote server that is configured to be visible to others.

See also

“Open a project” on page 96

“Use the Adobe dialog box” on page 86

Open a project file

1 In Acrobat, Flash, Illustrator, InCopy, InDesign, or Photoshop, choose File > Open and click Version Cue in the Favorites panel of the Adobe dialog box.

2 Double-click the project that contains the file you want to open.

3 Select the file, and click Open.

***Note:** To reveal a file in Bridge, right-click (Windows) or Control-click (Mac OS) the file in the Adobe dialog box, and choose Reveal In Bridge.*

Open a project file from Bridge

1 Click Version Cue in the Favorites panel.

2 Double-click the project that contains the file you want to open, and then double-click the file. The file opens in its native application.

Save changes to a local project file

If you want to save changes, but aren't ready to save a new version as you edit a file you have opened from a Version Cue project, you can use the File > Save command to save your changes to the local project file on your computer. Until you save a new version to the shared Version Cue Server, these changes won’t be available to any other user. You can also close the file once you save changes, and then reopen the file and check in a version later.

❖ To save changes to your local project file, choose File > Save.

See also

“Accessing Version Cue features” on page 85

“Check in versions” on page 108

Remove local project files

***Note:** You can perform this task only if you have access to the full Version Cue feature set. See "Accessing Version Cue features" on page 85.*

You can remove files that are not checked out by you from your local project folder if you want to free up more space on your hard drive, for example. Removing local project files does not affect checked in files that are stored on the Version Cue Server. Version Cue creates new local project files the next time you synchronize the project.

Disconnecting from a project also removes local project files; however, the Disconnect command also removes shortcuts to the project from Bridge and the Adobe dialog box.

- In Bridge, select a Version Cue project or project file, and choose Tools > Version Cue > Clear Local Files.
- In Acrobat, Flash, Illustrator, InCopy, InDesign, or Photoshop, select a Version Cue project or project file, and choose Clear Local Files from the Tools or Project Tools (Acrobat) menu.

See also

"Disconnect from projects" on page 100

Edit files checked out by another user

***Note:** You can perform this task only if you have access to the full Version Cue feature set. See "Accessing Version Cue features" on page 85.*

If someone is editing a local copy of a master file, Version Cue changes the file's status to Checked Out or In Use (Acrobat), informs you that the file is already checked out when you try to edit your local project file, and allows you to decide whether or not to continue working with the file.

When finished with the file, both users can save a new version of the file to the Version Cue Server. Version Cue alerts all current users of the file about the presence of a new version in the Version Cue Server and gives them the option of downloading the latest version or continuing their edits.

Use Version Cue Server Administration to assign lock protection to a Version Cue project. Only the first user to edit an available file in a lock-protected project can check in a version of that file to the Version Cue project. For more information, see "Create and manage projects in Version Cue Server Administration" on page 118.

Edit a file checked out by another user

1 Open the file, and choose one of the following options when the Checked Out By or In Use By (Acrobat) alert appears:

Discard Changes Displays the most recent version of the file from the Version Cue Server and discards your changes to the local project file.

Continue Editing Lets you edit the local project file without overwriting the changes made in another user's local copy of the same file (Version Cue will prompt each user to save a new version of the file).

2 If you continue working with the document and make a change to the content, Version Cue displays an alert to remind you that there is the possibility of creating conflicting copies. Choose one of the following:

No, Close Document Closes the file without any alterations.

Yes, Keep Open Keeps the file open so that you can work on the document.

3 If the project doesn’t have lock protection applied to it, you can save a new version of your edits. Version Cue displays an alert, warning you that conflicting edits will occur if you continue. Choose one of the following:

Cancel Returns you to the open document without checking in a version.

Check In Updates the master file in the Version Cue Server with the new version. (Version Cue displays an alert to the other user to note that a newer version of the file has been created.)

At any point, you can close the document and discard any changes you made.

Update a file with the most recent version

If another user creates a new version of a file that you have open or that is still marked as Checked Out or In Use (Acrobat), Version Cue prompts you to update your document with the latest version when you open it or attempt to make changes to it, or when you bring the document window frontmost in a group of documents.

❖ When the prompt appears, choose one of the following:

Discard Changes Updates the document with the most recent version from the Version Cue project. You can continue editing the file after it is updated. You lose any changes you made even if you have already used the Save command to save those changes to the local project file.

Continue Editing Leaves the document as is. You can continue editing the file without overwriting the changes in the more recent version. Instead, you’re prompted either to save a new version of the file when you close it or to discard your changes.

Move and copy Version Cue files

***Note:** You can perform this task only if you have access to the full Version Cue feature set. See “Accessing Version Cue features” on page 85.*

Use the Move To or Copy To commands in Bridge to move or copy Version Cue files within a project, among projects, or from a project to a desktop folder. When you copy or move a file, Version Cue copies or moves only the most current version.

See also

“Open a project” on page 96

“Use the Adobe dialog box” on page 86

Copy Version Cue files

❖ Do any of the following:

- Select the file in Bridge and choose Edit > Copy.
- Right-click the file in Bridge, choose Copy To, and choose a project or folder from the context menu (to specify a folder not listed, choose Folder, select a desktop or project folder, and click OK).
- Ctrl-drag (Windows) or Option-drag (Mac OS) the files to a different location.
- Drag the files from one project to another (if you drag the files to a different location in the same project, they are moved).

Move Version Cue files

❖ Do any of the following:

- Right-click the file in Bridge, choose Move To, and choose a project or project folder from the context menu (to specify a folder not listed, choose Folder, select a folder, and click OK).

Note: *Bridge moves files if you use the Move To command within the same Version Cue project. If you use the Move To command to move files from one Version Cue project to another or from a Version Cue project to a desktop folder, Bridge copies the files.*

- Drag a file to a different location in the same project (if you drag the files from one project to another, they are copied).

Search for Version Cue files

Note: *You can perform this task only if you have access to the full Version Cue feature set. See "Accessing Version Cue features" on page 85.*

You can locate files in a Version Cue project by searching for specific metadata such as titles, authors, copyright data, keywords, dates, and locations. Metadata is added to Version Cue project files as you work with them. In addition, you can add other metadata to files in Adobe Creative Suite components through the File Info dialog box. Adobe Creative Suite components can contain specific metadata fields; for example, fonts and colors in InDesign files and colors in Illustrator files.

You can search for files deleted from projects as well as existing files. In Bridge, you can search for Version Cue project files by version comment and past versions. (For instructions on searching in Bridge, see "Search for files and folders" in Bridge Help.)

Note: *Bridge doesn't search metadata for Version Cue version comments unless you choose Checkin Comment from the Criteria menu in the Find dialog box.*

1 In Acrobat, Flash, Illustrator, InCopy, InDesign, or Photoshop, click Version Cue in the Favorites panel of the Adobe dialog box and double-click the project you want to search.

2 Click Project Search .

3 Choose an option from the Search Category menu and enter criteria in the adjacent box.

4 Click Search.

See also

"Accessing Version Cue features" on page 85

"Open a project" on page 96

"View, promote, and delete versions" on page 109

Placing Version Cue files

Note: *You can perform this task only if you have access to the full Version Cue feature set. See "Accessing Version Cue features" on page 85.*

While you're working with a Version Cue project in Illustrator, InCopy, InDesign, or Photoshop, you can add a Version Cue file to a document just as you would place a non-Version Cue file—by using the Place command. You can also drag a file from a Version Cue project in Bridge to an open Flash, Illustrator, InCopy, InDesign, or Photoshop file.

Always add assets to a Version Cue project before placing them in a Version Cue-managed file. When you place a file that is not managed by Version Cue into a file that is, you cannot keep track of the placed asset's versions or status.

The Links panel (in Illustrator, InCopy, and InDesign) displays additional information about placed files from Version Cue projects, identifying whether a linked file is being edited and which user is doing the editing. You can also use the Links panel to determine whether the linked file needs to be updated to a newer version from the Version Cue Server.

For complete information on placing files into documents, see the specific application's Help.

Replace a placed file with a previous version

You may find yourself working with multiple versions of a document that includes a link to a file with multiple versions. If you decide to promote an earlier version of the document that includes a link to an earlier version of the file, Version Cue links to the current version of the file in the promoted document.

For example, suppose you create an InDesign document, place a Photoshop file in the document, and create multiple versions of both the InDesign document and the Photoshop file. If you then decide to promote an earlier version of the InDesign document, the link to the placed Photoshop file points to the current version of the file—not the version of the file to which you originally linked in the promoted InDesign document.To resolve this, replace the linked file with a previous version.

Note: *Illustrator, InCopy, or InDesign may display a thumbnail of the version of the file to which you originally linked in the promoted document, but the link actually points to the most current version of the file. For example, when you package an InDesign document that displays a thumbnail of the correct version, InDesign replaces the thumbnail of the file with the most current (but incorrect) version.*

1 In Illustrator, InCopy, or InDesign, select the file in the Links panel.

2 Choose Versions from the Links panel menu.

3 Select a version and click Promote To Current. Enter a version comment if desired, and click Save.

View files and versions in the Links panel

When Version Cue is enabled in Illustrator, InCopy, or InDesign, the Links panel identifies who is editing a linked file from a Version Cue project.

The Links panel functions the same with files that are managed with Version Cue as with files that are not. For example, if a newer version of a linked file is on the Version Cue Server, the Modified Artwork icon ⚠ appears; if a file is missing, the Missing Artwork icon ? appears. To update a linked file from a Version Cue project, you use the same procedures used for files that aren't managed by Version Cue.

The Links panel also displays a Version Cue status icon that describes the state of the file on the Version Cue Server (see "File statuses" on page 101), and displays a linked file's versions so you can promote and use previous versions. You can even create versions of linked non-Adobe files.

❖ Do one of the following:

- To view versions of a placed file, choose Versions from the Links panel menu.
- To view a tool tip that displays the versions of a placed file, place the pointer over the name of the file in the Links panel.

See Illustrator Help, InCopy Help, or InDesign Help for more information about working with the Links panel and placed files.

Delete files or folders from a project

***Note:** You can perform this task only if you have access to the full Version Cue feature set. See "Accessing Version Cue features" on page 85.*

You can delete files or folders in Bridge or any Version Cue-enabled Creative Suite component. *Deleting* a file or folder from Version Cue is a two-step process that safeguards against accidental deletions.

The first step is deleting the file or folder and giving it Deleted status. Deleting hides the file or folder from normal view but does not erase it. The second step is permanently deleting and erasing the file or folder and its previous versions.

***Note:** Any user with appropriate privileges can delete files and folders unless the files or folders are marked as Checked Out or In Use (Acrobat). If you're in a workgroup and a user is editing a file that you need to delete, you can reset the file's lock by using Version Cue Server Administration.*

In Bridge and Version Cue-enabled components of Adobe Creative Suite, you can view project files with Deleted status in Project Trash view. You can restore files or folders that have Deleted status to reinstate Version Cue management. Restored files and folders appear in their previous location in the project folder hierarchy.

See also

"Use the Adobe dialog box" on page 86

"Open a project" on page 96

"About local project files" on page 100

"Delete projects" on page 99

"Advanced Version Cue Server Administration tasks" on page 122

Delete files or folders

1 In Acrobat, Flash, Illustrator, InCopy, InDesign, or Photoshop, click Version Cue in the Favorites panel in the Adobe dialog box, and open the project that contains the files or folders you want to delete.

2 Select the file or folder you want to delete and click the Delete 🗑 icon in the toolbar.

Delete files or folders in Bridge

1 In Bridge, click Version Cue in the Favorites panel and open the project that contains the files you want to delete.

2 Select the file and click the Delete 🗑 icon in the toolbar.

Restore a deleted file or folder

1 In Acrobat, Flash, Illustrator, InCopy, InDesign, or Photoshop, open the project containing the file or folder that you want to restore in the Adobe dialog box.

2 Click Project Trash in the Favorites panel, right-click the file you want to restore, and choose Restore.

3 Choose Refresh from the Tools or Project Tools (Acrobat) menu to update the dialog box.

The file or folder is restored to its original location in the Version Cue project.

***Note:** To restore a file in a previously deleted folder, you must first restore the folder. Doing so restores the folder and all its contents.*

Restore a deleted file or folder in Bridge

1 Click Version Cue in the Favorites panel and open the project that contains the files you want to restore.

2 Choose Tools > Version Cue > View Project Trash.

3 Select the file you want to restore, and choose Tools > Version Cue > Restore.

The file or folder is restored to its original location in the Version Cue project.

Delete a file or folder permanently

1 In Acrobat, Flash, Illustrator, InCopy, InDesign, or Photoshop, open the project containing the file or folder that you want to delete permanently in the Adobe dialog box.

2 Click Project Trash in the Favorites panel, right-click the file you want to delete permanently, and choose Delete Permanent.

3 Click OK.

Delete a file permanently in Bridge

1 Click Version Cue in the Favorites panel and open the project that contains the files you want to delete permanently.

2 Choose Tools > Version Cue > View Project Trash.

3 Right-click (Windows) or Ctrl-click (Mac OS) the file you want to permanently delete, and click Delete Permanent.

Version Cue versions

About versions

Versions track changes to a file: each version is a snapshot of the file at a particular point in time. When you edit a file from the Version Cue Server, you're editing the last version saved to the Version Cue Server. When you're ready to save changes to the Version Cue Server, you check in a version. You don't have to check in a version every time you save your changes: check in a version only when you want to create a snapshot of the file.

You can save comments with versions to help you track changes. You can also promote a previous version to be the current version, letting you recover from unwanted changes.

You can compare multiple versions of the same file, and delete versions as they become obsolete or to save disk space.

For a video on managing versions, see www.adobe.com/go/vid0115.

See also

"Accessing Version Cue features" on page 85

"About local project files" on page 100

"Save changes to a local project file" on page 102

Check in versions

Note: *You can perform this task only if you have access to the full Version Cue feature set. See "Accessing Version Cue features" on page 85.*

To check in a new version of a file, use the Check In or Save A Version (Acrobat) command, which saves your changes to the Version Cue Server and removes the Checked Out or In Use By (Acrobat) status from the file.

When you check in a version, Version Cue transfers and saves only the changes you've made to the file.

You can check in versions of non-Adobe files only if the files are in a Version Cue project and the files have been opened through Bridge. After you check in versions of non-Adobe files, you can access them from the Versions dialog box in Version Cue-enabled Creative Suite components and from the Inspector or Content panel in Bridge.

***Note:** You can save versions of nonembedded graphics, image, and text files in InCopy, InDesign, and Illustrator by using the Edit Original command in the Links panel. After editing the file, save it in its native application. Then, in the Links panel, select the file and use the Save Link Version command to check in a version in the Version Cue project. For more information, see InCopy Help, InDesign Help, or Illustrator Help.*

See also

"Open a project" on page 96

Check in a version

1 Do one of the following:

- In Flash, InCopy, Illustrator, InDesign, or Photoshop, choose File > Check In.
- In Acrobat, choose File > Save A Version.
- In Bridge, select the file or files you want to check in and click the Check In button.

2 In the Check In dialog box, enter comments that you want to associate with the version, and then click OK.

Check in a non-Adobe file

1 Start Bridge.

2 In Bridge, click Version Cue in the Favorites panel and then open the project containing the file that you want to check out.

3 Double-click the file to check it out and open it.

4 When the file opens in its native application, make your changes, and save and close the file.

5 In Bridge, click the Check In button.

6 In the Check In dialog box, enter comments that you want to associate with the version, and then click OK.

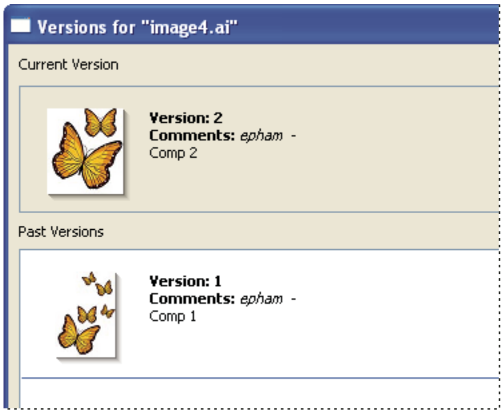
View, promote, and delete versions

***Note:** You can perform this task only if you have access to the full Version Cue feature set. See "Accessing Version Cue features" on page 85.*

Versions are treated as separate files, which you can access through the Versions dialog box in all Version Cue-enabled Creative Suite components or through the Content panel in Bridge. The Versions dialog box and Bridge display thumbnails of all file versions (numbered sequentially) with comments, dates, and the login name of the user who created the version.

If you want to compare versions in detail, you can choose to view each version in its native application.

If you want both a previous version and the current version to be available for simultaneous use in a project, save the previous version as a separate asset.



Versions for "image4.ai"

Current Version

Version: 2
Comments: *epham* -
Comp 2

Past Versions

Version: 1
Comments: *epham* -
Comp 1

Viewing versions in the Adobe dialog box

See also

"Use the Adobe dialog box" on page 86

"Open a project" on page 96

"Advanced Version Cue Server Administration tasks" on page 122

View versions

1 In Acrobat, Flash, InCopy, Illustrator, InDesign, or Photoshop, click Version Cue in the Favorites panel of the Adobe dialog box and open the project that contains the file whose versions you want to view.

2 Select the file whose versions you want to view.

3 Choose Versions from the Tools or Project Tools (Acrobat) menu .

Versions appear in the Versions dialog box.

You can view versions of a file while it's open in a Version Cue-enabled Creative Suite component: Choose Versions from the status menu at the bottom of the document window.

View versions in Bridge

1 Click Version Cue in the Favorites panel.

2 Open the project containing the file for which you want to view versions, and select the file.

3 Choose Tools > Version Cue > Versions, or click the Versions button in the Content panel. Versions appear in the Content panel.

View a previous version in its native application

If you view a previous version in its native application, changes you make won't be reflected in the current version (unless you promote the previous version to be the current version). You can, however, save edits to a previous version as a new asset.

❖ Do one of the following:

- In the Versions dialog box, click the version that you want to open and click View.
- In Bridge, double-click the version you want to open.

Version Cue opens the previous version in its native application. The version number appears in the file's title bar to remind you that it is not the current version. The file status is Never Saved, because the previous version is only a snapshot of a previous stage of the file.

Promote a version

Promoting a previous version saves a copy of the previous version as the current version. This process keeps the previous version intact, should you decide to return to it again in the future. Any changes made between its creation and promotion don't appear in the new current version.

1 Do one of the following:

- In the Versions dialog box, select the version you want to promote, and click Promote To Current Version.
- In Bridge, select the version you want to promote, and click Promote.

2 Type a version comment in the Check In dialog box and click Continue.

Delete a version

❖ Do one of the following:

- In the Versions dialog box, select the version you want to delete and click Delete.
- In Bridge, select the version you want to delete, and click Delete This Version.

Note that the remaining versions are not renumbered.

Using Version Cue Server Administration, you can delete multiple previous versions of all files in a project simultaneously. By using this method, you can retain past versions by date or by number of versions to keep. See "Advanced Version Cue Server Administration tasks" on page 122.

Editing and synchronizing offline files

About offline files

When you need to work on files from a Version Cue project while the Version Cue Server is unavailable, you can edit local project files on your computer. When the Version Cue Server is available again, you must *synchronize* your files with the Version Cue Server to save your latest version to the Version Cue Server. You can synchronize an entire project, or just a folder or a file in the project.

To prepare to work with offline files, it's best to first synchronize the entire Version Cue project while the server is still online to ensure that you have local project files. You can then edit the offline files and synchronize them once the server is back online.

Edit offline files

***Note:** You can perform this task only if you have access to the full Version Cue feature set. See "Accessing Version Cue features" on page 85.*

You can edit offline files from an unavailable Version Cue Server by opening offline copies. If you know that you'll be working with an offline file, you should first check the file out before going offline (see "Manually check out a file," below).

See also

"Use the Adobe dialog box" on page 86

"Open a project" on page 96

Edit local project files from an offline project

1 In Acrobat, Flash, InCopy, Illustrator, InDesign, or Photoshop, click Version Cue in the Favorites panel of the Adobe dialog box and open the project that contains the file you want to edit. It may take Version Cue a few seconds to verify that the Version Cue Server is unavailable.

2 Double-click the file to open it (the Offline Copy status allows you to open the file).

3 When you finish editing the file, choose File > Save to save the changes to the local project file. When the Version Cue Server becomes available again, synchronize your files. If the Version Cue Server becomes available while you're editing an offline file in an Adobe application, Version Cue will automatically mark the file as Checked Out or In Use (Acrobat).

Manually check out a file

If you intend to work on a file from an offline Version Cue Server, you should first manually mark the file as Checked Out before the server goes offline. When you mark a file as Checked Out, Version Cue creates a local project file for you and protects the file from editing by other users.

❖ Do one of the following:

- In Bridge, click Version Cue in the Favorites panel, navigate to the file, and click the Check Out button.
- In Acrobat, Flash, Illustrator, InCopy, InDesign, or Photoshop, click Version Cue in the Favorites panel of the Adobe dialog box and open the project that contains the file. Right-click (Windows) or Control-click (Mac OS) the file, and then choose Check Out or Mark In Use (Acrobat). Click Cancel to close the Adobe dialog box.

Synchronize files

***Note:** You can perform this task only if you have access to the full Version Cue feature set. See "Accessing Version Cue features" on page 85.*

If you've worked with offline files, you must *synchronize* your files with the Version Cue Server to save your latest version to the Version Cue Server. You may also want to synchronize if another user has saved a newer version of a file to the Version Cue Server.

See also

"Use the Adobe dialog box" on page 86

"Open a project" on page 96

Synchronize files

1 In Acrobat, Flash, Illustrator, InCopy, InDesign, or Photoshop, click Version Cue in the Favorites panel of the Adobe dialog box and select the project, folder, or file you want to synchronize. Do one of the following:

- To download assets from the Version Cue Server for which you have no corresponding local project files, choose Download from the Tools or Project Tools (Acrobat) menu (if Download is not available, the assets are already synchronized).
- To both upload and download assets to and from the Version Cue Server, choose Synchronize from the Tools or Project Tools (Acrobat) menu .

2 If prompted, choose an option in the File Conflict dialog box.

Synchronize files in Bridge

1 In Bridge, click Version Cue in the Favorites panel.

2 Select a project, folder, or file, and do one of the following:

- To download assets from the Version Cue Server for which you have no corresponding local project files, choose Tools > Version Cue > Download (if Download is not available, the assets are already synchronized).
- To both upload and download assets to and from the Version Cue Server, choose Tools > Version Cue > Synchronize (or click the Synchronize button in the toolbar).

3 If prompted, choose an option in the File Conflict dialog box.

File conflict options

If the master file on the Version Cue Server is newer than your local project file and you've made changes to the local project file, a File Conflict dialog box appears with the following options:

Apply The Following Action To All Subsequent Conflicts Automatically applies the selected option every time there is a file conflict.

Check In Saves your local project file as a new version to the Version Cue Server.

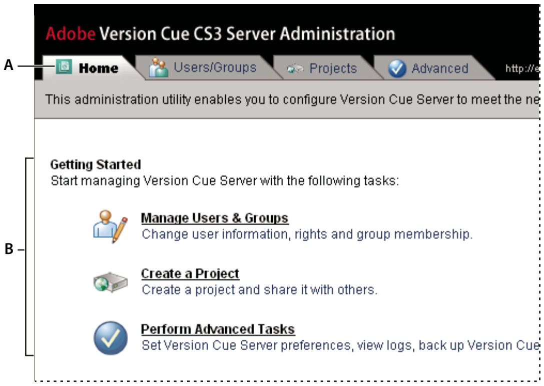
Skip This File Prevents the most recent version from the Version Cue Server from being downloaded. (This option also prevents a version of your local project file from being saved to the server.) Choose this option only if you want to keep your edits and disregard the other changes in the master file.

Version Cue Server Administration

About Version Cue Server Administration

Use Version Cue Server Administration to create, edit, and delete projects; manage user and group access; view logs and reports; initiate and manage web-based PDF reviews; and perform advanced server administration tasks such as deleting file versions, removing file locks, configuring plug-ins, and backing up the Version Cue Server.

The Version Cue Server Administration web page is divided into four tabs. Each tab contains controls that enable you to configure Version Cue. You can access Version Cue Server Administration from the Version Cue icon, from a web browser, or from any Version Cue-enabled Creative Suite component.



This administration utility enables you to configure Version Cue Server to meet the ne

Getting Started
Start managing Version Cue Server with the following tasks:

Manage Users & Groups
Change user information, rights and group membership.

Create a Project
Create a project and share it with others.

Perform Advanced Tasks
Set Version Cue Server preferences, view logs, back up Version Cue

Users/Groups and Projects tabs of Version Cue Server Administration
A. *Tabs* ***B.*** *Controls*

Version Cue Server Administration software requirements

Version Cue Server Administration for Windows requires the Java Runtime Environment (JRE) 1.5 or later to import projects from folders. You can download the Java Runtime Environment from the Sun Microsystems Java website at www.java.com/en/download/manual.jsp.

Version Cue Server Administration for both Windows and Mac OS requires Adobe Flash Player 9 for user and group administration. When you first create users and groups, Version Cue will prompt you to install Flash Player.

Log in to Version Cue Server Administration

Note: *You can perform this task only if you have access to the full Version Cue feature set. See "Accessing Version Cue features" on page 85.*

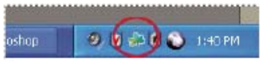
When you turn on the Version Cue Server for the first time, Version Cue automatically creates a default login name (**system**) with administrator privileges and asks you to specify a password. This login name and password let you log in to Version Cue Server Administration.

Other users with administrator privileges can also log in to Version Cue Server Administration.

Log in from the Version Cue icon

1 Do one of the following:

- (Windows) Right-click the Version Cue icon in the system tray and choose Server Administration.



On Windows, right-click the Version Cue icon.

- (Mac OS) Click the Version Cue icon at the top of the screen, and then click Server Administration.

2 Type your Version Cue login name and password in the boxes, and click Log In.

Log in from a Version Cue-enabled Creative Suite component

You can also log in to Version Cue Server Administration from Acrobat, Flash, InCopy, InDesign, Illustrator, and Photoshop.

1 Choose File > Open, and click Use Adobe Dialog.

2 Choose Connect To Server from the Tools menu or Connect To from the Project Tools menu (Acrobat), type the IP or DNS address and port of the Version Cue Server you want to administer, and click OK. The default port number is 3703 (50900 if you're connecting to a Version Cue CS3 server that's installed on the same system as a Version Cue CS2 workspace).

3 Choose Edit Properties from the Tools or Project Tools (Acrobat) menu .

4 Click Server Administration in the Edit Properties dialog box.

5 Type your Version Cue login name and password in the boxes, and click Log In.

Log in from a web browser

1 In a web browser, type the IP or DNS address of the computer on which the Version Cue Server is installed. Precede the address with http:// and follow it with a colon and the default port number, for example, http://153.32.235.230:3703 (IP) or http://*myserver.mycompany.com*:3703 (DNS). The default port number is 3703 (50900 if you're connecting to a Version Cue CS3 server that's installed on the same system as a Version Cue CS2 workspace).

Note: *If the server is installed locally, type http://localhost:3703.*

2 A browser window displays the Adobe Version Cue Server Administration login page. Type your Version Cue login name and password in the boxes, and click Log In.

Create and manage users

Note: *You can perform this task only if you have access to the full Version Cue feature set. See "Accessing Version Cue features" on page 85.*

Only users who have been granted System Administrator access privileges can create, import, export, and edit Version Cue users.

If you didn't enable automatic user account creation when you turned on the Version Cue Server, you need to create Version Cue user names to let other users access projects on the Version Cue Server. To restrict the Version Cue projects that a user can access, you can require login for the project and assign user names and permissions to that project.

Adobe Flash Player 9 is required to create and manage users in Version Cue Server Administration. When you first create users, Version Cue will prompt you to install Adobe Flash Player 9.

See also

"Create and manage projects in Version Cue Server Administration" on page 118

Create, edit, or delete users

Create users to let them access projects on the Version Cue Server.

1 Click the Users/Groups tab in Version Cue Server Administration, and then click New in the Users area.

2 In the New User dialog box, enter a user name, login, password, and choose the level of access to give the user from the Admin Access Level menu:

- None denies the user access to Version Cue Server Administration.
- User grants standard access to Version Cue Server Administration. Users with standard access can create new projects (if also granted project creation permissions) and modify projects they have created.
- System Administrator grants full access to all tasks in Version Cue Server Administration.

3 Select Project Creation to enable the user to create new Version Cue projects.

4 (Optional) Type a phone number, an e-mail address, and comments in the remaining boxes. Make sure to enter an e-mail address if the user will participate in Version Cue PDF reviews.

5 Click Save.

To edit a user, select the user, click Edit, change settings in the Edit [User Name] dialog box, and click Save. To delete a user, select the user, and click Delete.

Create, edit, or delete a user group

Create user groups to group users with similar permissions. For example, create a user group named “Designers” to group all users who are contributing artwork to a design project. The default group “Everyone” contains all users in the system.

1 Click the Users/Groups tab in Version Cue Server Administration.

2 Click New in the Groups area.

3 In the New Group dialog box, enter a name for the group. Optionally, enter a comment, and then click Save.

4 Add users to the group by dragging them from the Users area to the new group.

To change the name of a group, select it, click Edit, and enter a new name in the Groupname box. To delete a group, select it, and click Delete.

Assign permissions to users and groups

You can assign permissions to individual users, or to a group of users. Permissions are different from access levels: Access levels control access to Version Cue Server Administration, while permissions control access to the Version Cue Server, projects, and Version Cue PDF reviews. Note that permissions you assign to users or groups may be overwritten by permissions you assign to users for specific projects.

1 Do either of the following:

- To assign permissions to a user, select the user in the Users/Groups tab of Version Cue Server Administration.
- To assign permissions to all users in a group, select the group in the Users/Groups tab of Version Cue Server Administration.

2 Select Allow or Deny for each permissions category in the Global Permissions section:

To allow or deny all permissions, choose Allow or Deny from the Presets menu. To display the default permissions assigned to a user or group, select the user or group and click Effective Permissions.

- Read allows viewing projects and the files, versions, and file information within them.
- Write allows adding files to a project and saving versions and file information.
- Delete allows deleting projects or the files within them.
- Review Initiator allows initiating PDF reviews in Version Cue Server Administration (see “Start a Version Cue PDF review” on page 125).
- Project Administration allows administering projects (for example, duplicating, backing up, exporting, and deleting projects).

3 Click Save Permissions.

Import users from an LDAP directory

LDAP (Lightweight Directory Access Protocol) is a method of querying directory systems that contain information, such as user names and passwords, about users. You can import users from an LDAP server and map their user attributes (such as user name and password) to Version Cue user attributes. Users that you import from an LDAP server appear with a user icon that is different from the typical user icon.

Note: *If you enabled Automatic User Account Creation when you configured the Version Cue Server and then subsequently enable LDAP support, LDAP users are automatically imported when they access the server with their LDAP account name. Users imported in this fashion are added to the Everyone group, given a user access level of None, and are not able to log into Version Cue Server Administration. Use this technique to automatically assign LDAP users default access rights to projects on a Version Cue server without having to explicitly import users.*

1 Click the Advanced tab in Version Cue Server Administration.

2 Click LDAP Preferences.

3 Click Enable LDAP Support, and then enter information about the LDAP server:

- Enter the server name in the LDAP Server box.
- Enter the server port in the Server Port box.
- Enter the starting point in the LDAP hierarchy for the directory on the LDAP server in the Searchbase box.
- If the LDAP server requires authentication, enter a user name and password in the Username and Password boxes.
- Select Use LDAP with SSL if you want to connect via SSL to an SSL-enabled LDAP server.
- Enter LDAP attributes in the User-Id, Displayname, E-Mail, Info, and Phone boxes. Version Cue maps these to the corresponding Version Cue Server attributes.
- To specify that the Version Cue Server periodically synchronizes with the LDAP server, select Enable Automatic Synchronization and specify a synchronization period.

4 Click Save.

5 In the Users/Groups tab of Version Cue Server Administration, click Click To Maximize in the Users area.

6 Click Import External Users.

7 Type the first few letters of the LDAP user name or names you want to import in the External User dialog box. (Version Cue auto-completes the entry.)

8 Select the name or names, and click Add.

9 Repeat steps 7 and 8 until you've added all desired LDAP users, and then click Import User.

Export a list of users

To add a set of users to another Version Cue Server, export a list of users and then copy it to the UsersExport folder in the Version Cue application folder of the other computer with a Version Cue Server. You can then use the export list to import users.

1 Click the Users/Groups tab in Version Cue Server Administration.

2 Click To Maximize in the Users area.

3 Click Export Users.

4 Select the users you want to export (Shift-click to select contiguous users, Ctrl-click to select noncontiguous users).

5 Type a name for the list in the filename box. Optionally, type remarks in the Comments box.

6 Click Export.

The location of the user list appears under the Export Users heading. To import this list into another Version Cue Server, copy this file into the destination server's Data/UsersExport folder in the Version Cue application folder.

Import users from a list

1 Click the Users/Groups tab in Version Cue Server Administration and then click Import Users.

2 Click the user list that you want to import.

3 Select the check box next to each user name that you want to import, or select the check box next to the User Name column label to select all user names.

4 Click Next.

Create and manage projects in Version Cue Server Administration

***Note:** You can perform this task only if you have access to the full Version Cue feature set. See "Accessing Version Cue features" on page 85.*

You can create a new blank Version Cue project, a project from files in a folder on the computer where the Version Cue Server is installed, or a project from a WebDAV or FTP server. Once you've created a project, you can edit its properties in the Projects tab at any time.

Create a new Version Cue project

1 Click the Projects tab in Version Cue Server Administration, and then click New:

- Click Blank Project to create an empty Version Cue project.
- Click Import From Folder to create a project that contains files from a folder of files on the hard drive.
- Click Import From FTP Server or Import From WebDAV Server to import a website or to import files from a folder on an FTP or WebDAV server.

2 Type a project name in the New Project Name box.

3 Specify Version Cue project properties (see "Version Cue project properties" below).

4 Click Create (if you've created a new blank project) or Next (if you've created a project from a folder of files on an FTP or WebDAV server or on your hard drive).

5 If you chose to import a project from a folder, do the following, and then click Import:

- If the content you're importing is a website, select Import Folder As A Website.
- To specify the folder to import from, click Browse and select a folder.

***Note:** Don't navigate away from Version Cue Server Administration after you click Import. If you navigate away before all files have been imported into the project, Version Cue will create the project, but the project won't contain all files.*

6 If you chose to import a project from an FTP or WebDAV server, do the following, and then click Import:

- If the content you're importing is a website, select Import FTP Directory As A Website or Import WebDAV Directory As A Website.
- In the FTP Server or WebDAV Server box, specify the server from which to import files, and type the port number in the Port box.
- To specify a folder, click Browse and select a folder.
- If a user name and password are required to access the server, type them in the User Name and Password boxes.

- To use a proxy server to connect to the server, select Use Proxy.
- To use passive mode to connect to the server, select Use Passive Mode.

7 If you chose to require login for the project, click Assign Permissions and assign permissions to users (see "Assign user permissions," below).

Version Cue project properties

Specify these options when creating or editing Version Cue projects in Version Cue Server Administration:

Share This Project With Others Users can be on your subnet, or they can be given the Version Cue Server IP or DNS address and port number to gain access to the Version Cue Server.

Require Login For This Project Ensures that only users with a Version Cue login ID and password have access to the project.

Note: *If you select this option after other users have already accessed the project without being authenticated, those users can still access the project without logging in to it. Make sure that you change their privileges as needed in the project's list of assigned users.*

Enable Lock Protection For This Project Restricts file versioning to sequential versions. Only the first user to edit an available file in a lock-protected project can check in a version of that file to the Version Cue project. Other users can't check in a version until the first user saves a version and closes the file or reverts to the project version of the file and closes it—other users must save their changes as completely new files with their own version thread.

Maximize Compatibility With CS2 Applications And Acrobat 8 Creates a project that uses the Version Cue CS2 project structure so that Adobe Creative Suite 2 or Acrobat 8 users can work with Version Cue CS3 projects.

Comments Stores any remarks you type about the project.

Assign user permissions

If you chose to require login when creating a project, you need to assign permissions to users to define their access to the project.

1 In the Assign Permissions area of Version Cue Server Administration, select the user or the group that contains the users for which you want to assign permissions.

2 Select Allow or Deny for each permissions category in the Permissions for [user name] section:

To allow or deny all permissions, choose Allow or Deny from the Presets menu. To display the effective global and project permissions assigned to a user or group, select the user or group and click Effective Permissions.

- Read lets the user see files, versions, and file information in the project.
- Write lets the user create files, versions, and file information in the project.
- Delete lets the user delete files from the project.
- Review Initiator lets the user initiate PDF reviews in Version Cue Server Administration (see "Start a Version Cue PDF review" on page 125).
- Project Administration lets the user administer projects (for example, duplicating, backing up, exporting, and deleting projects).

3 Click Set Permissions.

Duplicate a Version Cue project

Duplicate a project to start a new project with the same users and privileges. Version Cue duplicates the folder hierarchy within the project structure.

1 Click the Projects tab in Version Cue Server Administration.

2 Select the check box next to the project you want to duplicate, and click Duplicate.

3 In the Duplicate Project page, type a unique name for the project.

4 Edit project properties, and click Duplicate.

Delete a Version Cue project

1 Click the Projects tab in Version Cue Server Administration, and do one of the following:

- To delete one or more projects, select the check box next to each project you want to delete.
- To delete all listed projects, select the check box next to the Project Name column label.

2 Click Delete. The Delete Project page appears.

3 Select User Locks Will Be Ignored to delete the project even if a user has files checked out.

4 Click Delete.

Export a Version Cue project to your computer or to an FTP or WebDAV server

You can export the most recent version of all project files from the Version Cue Server. Export if you want to move files from one host computer (or server) to another, create a package of the most recent files for output, or simply create an archive of the final versions. Version Cue still manages projects moved between computers.

Note: *If you want to move a project, first decide whether to back it up (so that all past versions are also moved) or to export it (so that only the current versions of project files are moved).*

1 Click the Projects tab in Version Cue Server Administration. Select the check box next to the project you want to export, and click Export.

2 In the Export Project page, choose a protocol by which to export the project.

3 Do one of the following:

- If you chose Export Project To Folder in step 2, specify the folder to which you want to export the project.
- If you chose Export Project To FTP Server or Export Project To WebDAV Server in step 2, specify the server address in the Server Address box, specify a folder in the Directory box, and enter a user name and password (if required). To use a proxy server to connect, select Use Proxy. If you are connecting to the server through a firewall, or if you specified a port other than 21, select Use Passive Mode. (This is an option only if you choose FTP in the Protocol menu.)

4 Click Export.

Back up and restore projects

Note: *You can perform this task only if you have access to the full Version Cue feature set. See "Accessing Version Cue features" on page 85.*

When you back up a Version Cue project, Version Cue Server Administration creates backups of all the information in a Version Cue project, including all versions of all files in the project. Use a project backup to move a project from one Version Cue Server to another while retaining all the versions of that project. You can restore a backup copy that represents a Version Cue project as it was on a specific date. Restored project backups do not replace the original Version Cue project.

You can customize a backup configuration for your projects in the Version Cue project preferences. You can back up a project using a new configuration or an existing configuration. A backup configuration includes the ability to schedule a recurring backup for the project.

By default, project backups are stored in the Program Files/Common Files/Adobe/Adobe Version Cue CS3/Server/Backups folder (Windows) or the Library/Application Support/Adobe/Adobe Version Cue CS3/Server/Backups folder (Mac OS).

Back up a Version Cue project

1 Click the Projects tab in Version Cue Server Administration.

2 Click the check box next to the project name, and then click Back Up.

3 In the Backup Name box, accept the backup name, or type a new name.

4 Choose the project components that you want to back up: Project Content (which is always selected) to back up files, Project File Versions to back up all versions of the files, Project Metadata to back up embedded information entered in Adobe Creative Suite components, and Users/User Assignments to back up information about the users and their project privileges.

5 Click Back Up.

Restore a Version Cue project backup copy

1 Click the Projects tab in Version Cue Server Administration.

2 Click Project Backups.

3 Select the backup that you want to restore.

4 In the New Project Name box, type a name that is different from those of other projects in the Version Cue Server.

5 Do any of the following, and then click Restore:

- To retain the list of users that were assigned to the project, select Restore Users.
- To retain the same privileges for each assigned user, select Restore User Assignments.
- To add remarks, type them in the Comments box.

Create a new backup configuration

When you create a new configuration, it becomes the default for the project.

1 Click the Projects tab in Version Cue Server Administration.

2 Click the project for which you want to create a new backup configuration.

3 Click Backup Configurations, and then click New.

4 Type a name for the backup configuration in the Backup Name box.

5 Select what to back up in the Include list of options: Project Content (which is always selected) to back up files, Project File Versions to back up all the versions of the project, Project Metadata to back up embedded information entered in Adobe Creative Suite components, and Users/User Assignments to back up information about the users and their project privileges.

6 (Optional) Add remarks to the backup file in the Comments box.

7 Click Schedule, and choose an option from the Repeat menu if you want backups to occur automatically (choose Don't Repeat if you want to back up the project manually).

8 Click Save.

Advanced Version Cue Server Administration tasks

***Note:** You can perform this task only if you have access to the full Version Cue feature set. See "Accessing Version Cue features" on page 85.*

Perform advanced Version Cue Server Administration tasks, such as backing up the server, specifying proxies, and enabling SSL, in the Advanced tab of Version Cue Server Administration.

See also

"Change the location of the Data folder" on page 92

View Version Cue Server and plug-in information and log files

You can display the Version Cue Server version, name, Java version, database version, Version Cue URL (IP or DNS address), and WebDAV URL with Version Cue Server Administration.

You can also view the Version Cue Server log file, which tracks all server operations according to the level of detail you specify. Log files are saved in the Logs folder in the Version Cue application folder.

❖ Click the Advanced tab of Version Cue Server Administration and do any of the following:

- To view Version Cue Server information, click Server Info.
- To view information about installed Version Cue plug-ins, click Plugins Overview.
- To view the Version Cue Server log file, click Server Log.
- To specify the log level (Error, Warning, or Info), specify the maximum log size, or reduce the log size by saving it as a compressed file, click Preferences and set these options.

View a Version Cue import or export report

1 Click the Advanced tab in Version Cue Server Administration, and then click Reports.

2 Choose the type of report you want to view from the Report menu.

3 To view available reports from a single project, choose the project name from the Filter By menu. To view available reports from all projects on the Version Cue Server, choose All.

4 Click the project's name in the Project Name column to display the report.

5 To print a copy of the report, click Print View.

6 To return to the report list, click File List.

To delete a report, select it in the Report List and click Delete.

Back up the Version Cue Server

You can back up the complete Version Cue Server to move a complete server from one computer to another.

***Important:** If you restore a backup copy of the Version Cue Server, all current data on the server, including Version Cue projects, files, and versions, is replaced by the backup.*

Server backup files are saved to the default Backups folder in the Version Cue application folder.

1 Click the Advanced tab in Version Cue Server Administration, and then click Back Up Version Cue Data.

2 To add remarks about the server backup, type them in the Comments box.

3 Click Save. After the backup is complete, click OK to view the list of server backups.

Replace a project with a previous backup

To replace current projects on a Version Cue Server with a previous version, you first restore the backup. When you do this, Version Cue Server Administration shuts down.

1 Click the Advanced tab in Version Cue Server Administration, and then click Administer Backups.

2 Click the backup you want to restore and then click Restore. The Version Cue Server shuts down. Close the browser. (Notice that the Version Cue icon in the system tray indicates that it’s off .)

3 Start the Version Cue Server.

4 log in to Version Cue Server Administration.

Change the name of the Version Cue Server

1 Click the Advanced tab in Version Cue Server Administration, and then click Preferences.

2 Type a name in the Server Name box.

Specify HTTP and FTP proxies

1 Click the Advanced tab in Version Cue Server Administration, and then click Preferences.

2 Specify the default FTP proxy server for users importing projects from or exporting projects to an FTP server.

3 Specify the default HTTP Proxy server for users importing projects from or exporting projects to a WebDAV server.

Remove file locks from a Version Cue project

Remove file locks to remove the Checked Out or In Use (Acrobat) status of files designated as such. A user with system administrator access or with project-specific Project Administration privileges can remove file locks.

1 Click the Advanced tab in Version Cue Server Administration, and then click Reset Locks (under Maintenance).

- Choose a project from the Project Name menu.
- Choose a user from the User Name menu.

2 Click Reset Locks to remove the specified file locks.

Delete file versions in a project

Delete file versions to improve performance. Each time you check in a version, it’s stored in the Version Cue Server database. This database creates a file version history that lets you quickly return to any former state of the file. An extensive history takes up disk space and can degrade the performance of the Version Cue Server.

1 Click the Advanced tab in Version Cue Server Administration, and then click Remove Old Versions.

2 Choose a project from the Project Name menu.

3 To delete versions, select Delete All Versions Older Than, and then choose a month, day, and year.

4 To specify the maximum number of versions to remain in the server after you click Delete, select Number Of Versions To Keep, and then type a number in the box.

5 Click Delete.

Grant access to the server without an existing user account

If you select this option, Version Cue creates a new user account without a password when a new user accesses the Version Cue Server.

1 Click the Advanced tab in Version Cue Server Administration, and then click Preferences.

2 Select Automatic User Creation to enable users to access the server without an existing user account.

Enable SSL

Enabling Secure Sockets Layer (SSL) for the Version Cue Server enables secure communication between the server and Bridge or a Version Cue-enabled Creative Suite component. When you enable SSL, the Version Cue Server sends data over an encrypted connection.

***Note:** Acrobat 8 and Creative Suite 2 components can't connect to a Version Cue CS3 Server that uses SSL.*

1 Click the Advanced tab in Version Cue Server Administration, and then click Security Preferences.

- To enable SSL, select Use SSL.
- To view the existing SSL certificate, click View The Currently Installed SSL Certificate.
- To load a custom SSL certificate, click Import A Custom SSL Certificate, select the certificate you want to use, and click Import.

2 Click Save.

Restart the Version Cue Server

1 In the Advanced tab of Version Cue Server Administration, click Restart Server.

2 Click Restart.

You can also restart the Version Cue Server in Version Cue Preferences.

Version Cue PDF reviews

About Version Cue PDF reviews

Using Version Cue Server Administration, you can set up and conduct web-based reviews of PDF documents that are on a Version Cue Server.

You can conduct Version Cue PDF reviews for Adobe Illustrator (AI) files that have been saved with the Enable PDF Compatibility option without first converting them to PDF. The AI files appear with PDF files in the Document List when you start a Version Cue PDF review.

As the review progresses, reviewers upload their comments to the Version Cue Server. When a review is complete, you can view all comments either in the context of the original document or as a list in Version Cue Server Administration.

- To use Version Cue PDF review, reviewers need a Version Cue login name and privileges that allow them to log in to the Version Cue Server hosting the review.
- To view the PDF and add comments, users need Acrobat 7.0 Professional or later. For more information about commenting in Acrobat, see Acrobat Help.

Start a Version Cue PDF review

You can start a Version Cue PDF review for any version of any PDF document that is on a Version Cue Server, provided that you have appropriate privileges to access Version Cue Server Administration. Only one version of a PDF document may be in review at any point in time.

1 Log in to Version Cue Server Administration. (For instructions, see "Log in to Version Cue Server Administration" on page 114.)

2 Click the Version Cue CS3 PDF Review link at the top of the page.

- In the main Version Cue CS3 PDF review page, click Start A Review.
- Click the Documents tab, and choose Not Started from the Review Status menu.

3 In the Document List, click the name of the PDF document you want to review.

4 Choose the version you want to review, and then click Start Review.

5 In the Start Review page, enter review information:

- To set an end date for the review, select Deadline, and then choose the end date from the Year, Month, and Day menus.
- To let reviewers see each other's comments, select Open under Review Mode. Select Private if you want reviewers to see only their own comments.
- Type a description of the review in the Description box.
- To add reviewers, select the reviewers' names in the Reviewers section. (Click the check box next to the Reviewers column label to select or deselect all reviewers.)

Note: *If a reviewer is outside your workgroup and doesn't have a Version Cue login, you need to set one up in advance. You must also provide network access—typically through a firewall—for outside reviewers.*

6 Click Next.

7 To send an e-mail invitation to reviewers, select Send E-Mail Invitation, and then modify the Mail Subject and Mail Message as desired. In the E-Mail Recipients section, choose reviewers that you wish to invite by e-mail.

8 Click Start Review.

9 If you chose to invite reviewers by e-mail, Version Cue starts your e-mail program and displays an e-mail message addressed to the reviewers. The e-mail includes a direct link to the document being reviewed. Confirm the contents of the review e-mail, and send it.

Manage PDF reviews

After you locate a PDF review, you can open it, view or delete review comments, edit review settings, stop or restart a review, or delete the review from the Version Cue Server.

Locate PDF reviews

1 log in to Version Cue Server Administration. (For instructions, see "Log in to Version Cue Server Administration" on page 114.)

2 Click the Version Cue CS3 PDF Review link at the top of the page.

3 Do one of the following:

- If you don't know the name of the PDF document under review, or want to view all active reviews, click Active Reviews in the Home tab.
- If you don't know the name of the PDF document for which a review has been completed, or want to view all completed reviews, click Finished Reviews in the Home tab.
- If you want to search for a PDF document that is under review or for which a review has been completed, click Search Documents in the Home tab, and choose search criteria from the Project Name, Review Status, and List Entries menus. To find a PDF document by its name, enter the name or part of it in the Document Name field. Click Search.

Open an active or completed PDF review

1 Locate the review.

2 Click the PDF document name in the Document List, and then select any of the versions in the Document History list.

Stop a PDF review

1 Locate the review.

2 Click the PDF document name in the Document List.

3 In the Document History list, select the active review and click Stop Review.

To restart a completed review, click Start Review in the Document History list.

***Note:** After you click Start Review, you see a series of screens that refer to starting, rather than restarting, a review. However, this procedure does restart the review of the existing document.*

Delete a PDF review

When you delete a review, Version Cue permanently removes the review comments. However, review comments for a PDF file are also deleted if you permanently delete the file itself from the Version Cue Server. Note that if you delete only a version of a PDF file from the server, the review comments for that version are deleted.

1 Locate the review.

2 Click the PDF document name in the Document List.

3 In the Document History list, select a version and click Delete Review.

4 When Version Cue prompts you to delete the review, click Delete.

Edit review settings

1 Locate the review.

2 Click the PDF document name in the Document List.

3 Select one of the versions in the Document History list, and click Edit Review Settings.

- To set or change an end date for the review, select Deadline, and then choose the end date from the Year, Month, and Day menus.

- To let reviewers see each other’s comments, select Open under Review Mode. Select Private if you want reviewers to see only their own comments.
- To add or edit a description of the review, type the information in the Description box.
- To add or remove reviewers, select or deselect the reviewers’ names in the Reviewers section (click the check box next to the Reviewers column heading to select or deselect all reviewers).

4 Click Next.

5 To send an e-mail invitation to reviewers, select Send E-Mail Invitation, and then modify Mail Subject and Mail Message as desired. In the E-Mail Recipients section, choose reviewers that you wish to invite by e-mail.

6 Click Save Review. If you chose to invite reviewers by e-mail, Version Cue starts your e-mail program and displays an e-mail message addressed to the reviewers. This e-mail includes a direct link to the document being reviewed. Confirm the contents of the review e-mail, and send it.

Set viewing options in the Document List

- To display only PDF documents in a specific project, choose that project from the Project menu.
- To limit the number of documents displayed, choose an option from the List Entries menu (use the arrows to the right of the List Entries menu to view additional files).
- To limit the list according to document name, enter part of a document name in the Document Name field and press Enter (Windows) or Return (Mac OS). (To view all files again, delete the text in the Document Name field and press Enter or Return.)
- To sort the list by the entries in a column, click the column heading. (Click the heading again to reverse the sort order.)

View or delete PDF review comments

Review comments include, in addition to the text of the comment itself, information about who created the comment and when, what type of comment was created, and what page of the document the comment appears on. You can use any of the Acrobat commenting tools in a Version Cue PDF review.

Version Cue stores review comments on the Version Cue Server. You can view comments in Version Cue Server Administration or directly in the PDF document. To view all review comments directly in the document, you must access the document either by using the link from the review invitation or by opening the review document from Version Cue Server Administration. (If you open the review document from the Open dialog box in Acrobat or from Bridge, the review comments aren’t visible.)

For more information about Acrobat commenting tools, search for “commenting” in Acrobat Help.

1 Locate the review.

2 Click the PDF document in the Document List.

3 Do one of the following:

- To view all review comments directly in the PDF document, click the version name.
- To view review comments in Version Cue Server Administration, select the version in the Document History list and click View Comments.

To view any of the comments in the context of the PDF document, select a comment and then click Open In Acrobat.

- To delete review comments in Version Cue Server Administration, select the comment and click Delete Comments. (To select all comments, click the check box next to the Page column heading.)

See also

“Manage PDF reviews” on page 125

Troubleshooting

Multiple local project files folders

If you accessed a project that has the same name on two different Version Cue Servers, Version Cue may create multiple copies of what appear to be the project’s local project files folder, but are actually the local project files folders of the two different projects accessed from the different servers. Version Cue names these folders with the project name and a number (for example, project_001, project_002). You can delete these folders after you save versions of your local project files to the Version Cue Server. (Or, you can simply ignore the multiple local project file folders.)

To delete the local project files folders, use Adobe Bridge to disconnect from each project (see “Disconnect from projects” on page 100).

Offline projects

If your network access becomes unavailable while you’re working with a Version Cue project (for example, if you’re traveling with a laptop computer), Version Cue may show the project as offline the next time you connect. To reestablish a connection to a Version Cue Server, select the Version Cue icon in the Favorites panel of the Adobe dialog box or Adobe Bridge (this forces Version Cue to reset its network connection to available servers). Then, locate the Version Cue project that contains your files.

Incomplete actions

In some cases, less frequently used commands don’t automatically update the Version Cue information displayed in Adobe Bridge or in the Adobe dialog box. If you don’t see the results of an action you have completed, refresh the view by doing one of the following:

- Change the focus in the Bridge window or Adobe dialog box by selecting another item.
- Change to a different folder momentarily.
- Resize the Bridge window or the Adobe dialog box, or bring the Bridge window to the foreground (updates may not appear in the Bridge window if it’s in the background).
- Choose the Refresh command using the Tools menu or the Project Tools (Acrobat) menu in the Adobe dialog box or the View menu in Adobe Bridge.

Project recovery

In the very unlikely event that you’re completely unable to access a Version Cue project, you can recover the current versions of files found in the local project files folders on the computers of workgroup members who have accessed the project. Local project files are located in the Version Cue folder in your My Documents (Windows) or Documents (Mac OS) folder.

Files don’t appear in projects behind firewalls

Version Cue uses HTTP (Hypertext Transfer Protocol) and SOAP (Simple Object Access Protocol) to communicate between Version Cue Servers and Adobe Creative Suite components. Rarely, older firewall software may not handle SOAP interactions properly. If you see folders but not files in your Version Cue projects, try disabling your proxy server. If disabling the proxy server resolves the problem, you may need to update your firewall.

Disconnecting from projects with files checked out

If you have local project files with the Checked Out By Me status, you can’t disconnect from a project until you check in a version of those files. If you can’t check in a version of the files because the Version Cue Server is unavailable, there are two ways to proceed. You can wait until the server is available, synchronize the files, and then disconnect from the project. Or, you can delete the local project files folder from your hard drive and then disconnect from the project when the server is available. (Other users who access the project will still see the files marked as Checked Out. Users can save their own versions, or a user with Administrator privileges can reset locks on the project, which changes the status of the files to Synchronized.)

Unable to see thumbnails for InDesign files in the Adobe dialog box or Bridge

If you don’t see thumbnails for InDesign files in the Adobe dialog box or in Bridge, select Always Save Preview Images With Documents in either the File Handling preferences or the Save As dialog box in InDesign.

Unable to migrate from Version Cue CS2 to Version Cue CS3 on Intel-based Macintosh computers

To migrate projects from Version Cue CS2 to Version Cue CS3 on an Intel-based Macintosh computer, first back up the project in the Version Cue CS2 Advanced Administration utility on a PowerPC-based Macintosh computer. (For instructions, see Version Cue CS2 Help.) Then, transfer the project from the PowerPC-based Macintosh to the project backup directory on the Intel-based Macintosh computer, restart the server, and restore the project in Version Cue CS3 (see “Back up and restore projects” on page 120).

***Note:** You cannot migrate Version Cue CS2 server backups to a Version Cue CS3 Server. You can, however, migrate Version Cue CS2 projects to a Version Cue CS3 Server. See “Migrate projects to the Version Cue 3.0 Server” on page 93*

Unable to connect to the Version Cue CS3 Server

Try any of the following:

- Make sure that your network connection is functioning properly.
- If you’re trying to connect to the server from Acrobat 8 or from a Version Cue-enabled Adobe Creative Suite 2 component, make sure that you haven’t enabled SSL for the server. In addition, Acrobat 8 and Adobe Creative Suite 2 components can’t connect to Version Cue CS3 Servers installed on the same computer (that is, a local Version Cue CS3 Server).
- Try to connect to the server from a different Version Cue-enabled Creative Suite component.
- Temporarily disable firewalls or proxies.
- Restart the server (see “Shut down or restart the Version Cue Server” on page 94).
- Consult the server log file for details that may indicate the cause of the problem. You can access the log file from the Advanced tab of Version Cue Server Administration (see “Advanced Version Cue Server Administration tasks” on page 122).

Forgotten Version Cue system administrator password

If you've forgotten your system administrator password, you'll need to remove and reinstall Version Cue, creating a new system administrator user name and password in the process. There is no way to remove Version Cue without losing all Version Cue project files and data.

Version Cue Server performs slowly or stops responding

Try either of the following:

- Increase the RAM allocated to Version Cue (see "Set Version Cue Server preferences" on page 91).
- If Version Cue stops responding while you're backing up project files, make sure that you're backing up to a drive with enough free hard drive space to store the files.

Unable to see Version Cue projects on a Version Cue CS3 Server

Try any of the following:

- Make sure that the project to which you're trying to connect is shared. If the project creator chose to keep the project private, you won't be able to access it.
- Make sure that the Version Cue Server that hosts the project is configured to be visible to other users (see "Set Version Cue Server preferences" on page 91).
- If you're using Acrobat 8 or a Version Cue-enabled Adobe Creative Suite 2 component to access the project, make sure the creator of the project specified that it be compatible with CS2. Otherwise, you won't be able to see the project on the Version Cue CS3 Server.

Keyboard shortcuts

About keyboard shortcuts

Keyboard shortcuts let you quickly select tools and execute commands without using a menu. When available, the keyboard shortcut appears to the right of the command name in the menu. This is not a complete list of keyboard shortcuts. These tables list only those shortcuts that are not displayed in menu commands or tool tips.

In addition to using keyboard shortcuts, you can access many commands using context-sensitive menus. Context-sensitive menus display commands that are relevant to the active tool, selection, or palette. To display a context-sensitive menu, right-click (Windows) or Control-click (Mac OS) in the document window or palette.

Keys for selecting in Version Cue

Result	Windows	Mac OS
Select all	Control+A	Command+A
Select entries (selective)	Control-click	Command-click
Select entries (range)	Shift-click	Shift-click
Select next entry	Down Arrow	Down Arrow
Select previous entry	Up Arrow	Up Arrow
Select next entry (additive)	Shift + Down arrow	Shift + DownAarrow
Select previous entry (additive)	Shift + Up Arrow	Shift + Up Arrow
Select first entry	Page Up or Home	Page Up or Home

Result	**Windows**	**Mac OS**
Select last entry	Page Down or End	Page Down or End
Select an entry and all entries above	Shift + Page Up or Home	Shift + Page Up or Home
Select an entry and all entries below	Shift + Page Down or End	Shift + Page Down or End

Keys for navigation in Version Cue

Result	**Windows**	**Mac OS**
Open folder	Control + O	Command + O
New folder	Control + N	Command + N
Up one level	Backspace	Command + Up Arrow
Open selected file	Enter	Return
Delete selected file or files	Delete	Command + backspace
Cancel dialog	Escape	Escape
Go forward	Control + Right Arrow	Command + Right Arrow
Go back	Control + Left Arrow	Command + Left Arrow
Go to My Computer	Control + Shift + C	Command + Shift + C
Go to desktop	Control + Shift + D	Command + Shift + D
Go to network	Control + Shift + K	Command + Shift + K
Go to Version Cue	Control + Shift + V	Command + Shift + V

Keys for viewing in Version Cue

Result	**Windows**	**Mac OS**
Details	Control + 1	Command + 1
Icons	Control + 2	Command + 2
Thumbnails	Control + 3	Command + 3
Tiles	Control + 4	Command + 4

Chapter 5: Using imported artwork

A key advantage of Adobe® Flash® CS3 Professional is that you can import artwork created in other applications, and use those assets in your flash documents. You can import vector graphics and bitmap images in a variety of file formats, making Adobe® Flash® CS3 Professional a versatile media arts tool.

When you import a bitmap, you can apply compression and anti-aliasing, place the bitmap directly in an Adobe® Flash® CS3 Professional document, use the bitmap as a fill, edit the bitmap in an external editor, break the bitmap apart into pixels and edit it in Adobe® Flash® CS3 Professional, or convert the bitmap to vector artwork.

Placing artwork into Flash

About importing artwork into Flash

Adobe® Flash® CS3 Professional can use artwork created in other applications. You can import vector graphics and bitmaps in a variety of file formats. If you have QuickTime 4 or later installed on your system, you can import additional vector or bitmap file formats. You can import Adobe® FreeHand® files (version MX and earlier) and Adobe® Fireworks® PNG files directly into Flash, preserving attributes from those formats.

Graphic files that you import into Flash must be at least 2 pixels x 2 pixels in size.

To load JPEG files into a Flash SWF file during runtime, use the `loadMovie` action or method. For detailed information, see loadMovie (MovieClip.loadMovie method) in *ActionScript 2.0 Language Reference* or “Working with Movie Clips” in *Programming ActionScript 3.0*.

Flash imports vector graphics, bitmaps, and sequences of images as follows:

- When you import Adobe® Illustrator® and Adobe® Photoshop® files into Flash, you can specify import options that let you preserve most of your artwork's visual data, and the ability to maintain the editability of certain visual attributes with the Flash authoring environment.
- When you import vector images into Flash from FreeHand, select options for preserving FreeHand layers, pages, and text blocks.
- When you import PNG images from Fireworks, import files as editable objects to modify in Flash, or as flattened files to edit and update in Fireworks.
- Select options for preserving images, text, and guides.

***Note:** If you import a PNG file from Fireworks by cutting and pasting, the file is converted to a bitmap.*

- Vector images from SWF and Windows Metafile Format (WMF) files that you import directly into a Flash document (instead of into a library) are imported as a group in the current layer.
- Bitmaps (scanned photographs, BMP files) that you import directly into a Flash document are imported as single objects in the current layer. Flash preserves the transparency settings of imported bitmaps. Because importing a bitmap can increase the file size of a SWF file, consider compressing imported bitmaps.

***Note:** Bitmap transparency might not be preserved when bitmaps are imported by dragging and dropping from an application or desktop to Flash. To preserve transparency, use the File > Import To Stage or Import To Library command for importing.*

- Any sequence of images (for example, a PICT and BMP sequence) that you import directly into a Flash document is imported as successive keyframes of the current layer.

For a video tutorial about the Flash and Fireworks workflow, see www.adobe.com/go/vid0194.

See also

"Working with imported bitmaps" on page 155

"Working with video" on page 300

"Working with sound" on page 291

"Set bitmap properties" on page 155

Place artwork in Flash

Flash lets you import artwork in various file formats either directly to the stage, or to the library.

See also

"Using symbols, instances, and library assets" on page 207

Import a file into Flash

1 Do one of the following:

- To import a file directly into the current Flash document, select File > Import > Import To Stage.
- To import a file into the library for the current Flash document, select File > Import > Import To Library. (To use a library item in a document, drag it onto the Stage.

2 Select a file format from the Files Of Type (Windows) or Show (Macintosh) pop-up menu.

3 Navigate to the desired file and select it. If an imported file has multiple layers, Flash might create new layers (depending on the import file type). Any new layers appear in the Timeline.

4 Click Open.

5 If the name of the file you are importing ends with a number, and additional sequentially numbered files are in the same folder, do one of the following:

- To import all the sequential files, click Yes.
- To import only the specified file, click No.

The following are examples of filenames that can be used as a sequence:

Frame001.gif, Frame002.gif, Frame003.gif

Bird 1, Bird 2, Bird 3

Walk-001.ai, Walk-002.ai, Walk-003.ai

Paste a bitmap from another application directly into the current Flash document

1 Copy the image in the other application.

2 In Flash, select Edit > Paste In Center.

Importing file formats for vector or bitmap files

Flash can import different vector or bitmap file formats depending on whether QuickTime 4 or later is installed on your system. Using Flash with QuickTime 4 installed is especially useful for collaborative projects in which authors work on both Windows and Macintosh platforms. QuickTime 4 extends support for certain file formats (including PICT, QuickTime Movie, and others) to both platforms.

You can import the following vector or bitmap file formats into Flash 8 or later, regardless of whether QuickTime 4 is installed:

File type	Extension	Windows	Macintosh
Adobe Illustrator (version 10 or earlier	.ai	•	•
Adobe Photoshop;	.psd	•	•
AutoCAD DXF	.dxf	•	•
Bitmap	.bmp	•	•
Enhanced Windows Metafile	.emf	•	
FreeHand	.fh7, .fh8, .fh9, .fh10, .fh11	•	•
FutureSplash Player	.spl	•	•
GIF and animated GIF	.gif	•	•
JPEG	.jpg	•	•
PNG	.png	•	•
Flash Player 6/7	.swf	•	•
Windows Metafile	.wmf	•	•

You can import the following bitmap file formats into Flash only if QuickTime 4 or later is installed:

File type	Extension	Windows	Macintosh
MacPaint	.pntg	•	•
PICT	.pct, .pic	• (As bitmap)	•
QuickTime Image	.qtif	•	•
Silicon Graphics Image	.sgi	•	•
TGA	.tga	•	•
TIFF	.tif	•	•

Working with Illustrator and Flash

You can move Illustrator artwork into the Flash editing environment or directly into Flash Player. You can copy and paste artwork, save files in SWF format, or export artwork directly to Flash. In addition, Illustrator provides support for Flash dynamic text and movie clip symbols.

You can also use Device Central to see how Illustrator artwork will appear in Flash Player on different handheld devices.

Pasting Illustrator artwork

You can create graphically rich artwork in Illustrator and copy and paste it into Flash simply, quickly, and seamlessly.

When you paste Illustrator artwork into Flash, the following attributes are preserved:

- Paths and shapes
- Scalability
- Stroke weights
- Gradient definitions
- Text (including OpenType fonts)
- Linked images
- Symbols
- Blending modes

In addition, Illustrator and Flash support pasted artwork in the following ways:

- When you select entire top-level layers in Illustrator artwork and paste them into Flash, the layers are preserved along with their properties (visibility and locking).
- Non-RGB Illustrator colors (CMYK, grayscale, and custom) convert to RGB in Flash. RGB colors paste as expected.
- When you import or paste Illustrator artwork, you can use various options to preserve effects (such as drop shadow on text) as Flash filters.
- Flash preserves Illustrator masks.

Exporting SWF files from Illustrator

From Illustrator, you can export SWF files that match the quality and compression of SWF files exported from Flash.

When you export, you can choose from a variety of presets to ensure optimal output, and you can specify how to handle symbols, layers, text, and masks. For example, you can specify whether Illustrator symbols are exported as movie clips or graphics, or you can choose to create SWF symbols from Illustrator layers.

Importing Illustrator files into Flash

When you want to create complete layouts in Illustrator and then import them into Flash in one step, you can save your artwork in the native Illustrator format (AI) and import it, with high fidelity, into Flash using the File > Import To Stage or File > Import To Library commands in Flash.

When you import Illustrator artwork as an AI, EPS, or PDF file, Flash preserves the same attributes as for pasted Illustrator artwork.

In addition, when an imported Illustrator file contains layers, you can import them in any of the following ways:

- Convert Illustrator layers to Flash layers
- Convert Illustrator layers to Flash frames
- Convert each Illustrator layer to a Flash graphic symbol
- Covert all Illustrator layers to a single Flash layer

Symbol workflow

Symbol workflow in Illustrator is similar to symbol workflow in Flash.

Symbol creation When you create a symbol in Illustrator, the Symbol Options dialog box lets you name the symbol and set options specific to Flash: movie clip symbol type (which is the default for Flash symbols), Flash registration grid location, and 9-slice scaling guides. In addition, you can use many of the same symbol keyboard shortcuts in Illustrator and Flash (such as F8 to create a symbol).

Isolation mode for symbol editing In Illustrator, double-click a symbol to open it in isolation mode for easier editing. In isolation mode, only the symbol instance is editable—all other objects on the artboard are dimmed and unavailable. After you exit isolation mode, the symbol in the Symbols panel, and all instances of that symbol are updated accordingly. In Flash, symbol-editing mode and the Library panel work in a similar fashion.

Symbol properties and links Using the Symbols panel or the Control panel, you can easily assign names to symbol instances, break links between instances and symbols, swap a symbol instance with another symbol, or create a copy of the symbol. In Flash, the editing features in the Library panel work in a similar way.

Static, dynamic, and input text objects

When you bring static text from Illustrator into Flash, Flash converts the text to outlines. If you have a Smart Object workflow in Illustrator, you can take the text back to Illustrator for editing. In addition, you can set up your text in Illustrator as dynamic text. Dynamic text enables you to edit text content programmatically in Flash, and easily manage projects that require localization in multiple languages.

In Illustrator, you can specify individual text objects as static, dynamic, or input text. Dynamic text objects in Illustrator and Flash have similar properties. For example, both use kerning that affects all characters in a text block rather than individual characters, both anti-alias text the same way, and both can be linked to an external XML file containing text.

About importing Fireworks PNG files

Import Fireworks PNG files into Flash as flattened images or as editable objects. When you import a PNG file as a flattened image, the entire file (including any vector artwork) is rasterized, or converted to a bitmap image. When you import a PNG file as editable objects, vector artwork in the file is preserved in vector format. Choose to preserve placed bitmaps, text, filters (called effects in FireWorks) and guides in the PNG file when you import it as editable objects.

For a video tutorial about the Flash and Fireworks workflow, see www.adobe.com/go/vid0194.

Using Fireworks effects and blend modes in Flash

Flash 8 and later provides support for Fireworks filters, blend modes, and text. When you import Fireworks PNG files, you can retain many of the filters and blend modes applied to objects in Fireworks, and continue to modify those filters and blends using Flash 8 or later versions.

Flash only supports modifiable filters and blends for objects imported as text and movie clips. If an effect or blend mode is not supported, Flash 8 and later versions rasterizes or ignores it when it is imported.

To import a Fireworks PNG file that contains filters or blends that Flash does not support, rasterize the file during the import process. After this operation, you cannot edit the file.

For a video tutorial about the Flash and Fireworks workflow, see www.adobe.com/go/vid0194.

Import text from Fireworks into Flash

When you import text from Fireworks into Flash 8 or later, the text is imported with the default anti-alias setting of the current document.

If you import the PNG file as a flattened image, you can start Fireworks from Flash and edit the original PNG file (with vector data).

When you import multiple PNG files in a batch, you select import settings one time. Flash uses the same settings for all files in the batch.

Note: *To edit bitmap images in Flash convert the bitmap images to vector artwork or break apart the bitmap images.*

For a video tutorial about the Flash and Fireworks workflow, see www.adobe.com/go/vid0194.

1 Select File > Import > Import To Stage or Import To Library.

2 Select PNG Image from the Files Of Type (Windows) or Show (Macintosh) pop-up menu.

3 Navigate to a Fireworks PNG image and select it.

4 Click Open.

5 Select one of the following for Location:

Import All Page(s) into New Scence(s) Imports all pages in the PNG file as scenes within a movie clip, with all of its frames and layers intact inside the movie clip symbol. A new layer is created that uses the name of the Fireworks PNG file name. The first frame (page) of the PNG document is placed on a keyframe that starts at the last keyframe; all other frames (pages) follow.

Import One Page into Current Layer Imports the selected page (idenified in the Page Name pop-up menu) of the PNG file into the current Flash document in a single new layer as a movie clip. The contents of the selected page is imported as a movie clip, with all of its original layer and frame structure intact. If the page movieclip has frames inside it, each frame is a movieclip in itself.

Page Name Specifies the Fireworks page you want to import into the current scene.

6 Select one of the following for File Structure:

Import As Movie Clip And Retain Layers Imports the PNG file as a movie clip, with all of its frames and layers intact inside the movie clip symbol.

Import Page(s) as New Layer(s) Imports the PNG file into the current Flash document in a single new layer at the top of the stacking order. The Fireworks layers are flattened into the single layer. The Fireworks frames are contained in the new layer.

7 For Objects, select one of the following:

Rasterize If Necessary To Maintain Appearance Preserves Fireworks fills, strokes, and effects in Flash.

Keep All Paths Editable Keeps all objects as editable vector paths. Some Fireworks fills, strokes, and effects are lost on import.

8 For Text, select one of the following:

Rasterize If Necessary To Maintain Appearance Preserves Fireworks fills, strokes, and effects in text imported into Flash.

Keep All Paths Editable Keeps all text editable. Some Fireworks fills, strokes, and effects are lost on import.

9 To flatten the PNG file into a single bitmap image, select Import As A Single Flattened Bitmap. When this option is selected, all other options are dimmed.

10 Click OK.

See also

“Edit bitmaps in an external editor” on page 157

“Convert bitmaps to vector graphics” on page 158

“Break apart a bitmap” on page 157

Import FreeHand MX files

You can import FreeHand files in version 7 or later directly into Flash. FreeHand is a good choice for creating vector graphics for import into Flash, because you can preserve FreeHand layers, text blocks, library symbols, and pages, and choose a page range to import. If the imported FreeHand file is in CMYK color mode, Flash converts the file to RGB.

Keep the following guidelines in mind when importing FreeHand files:

- When importing a file with overlapping objects to preserve as separate objects, place the objects on separate layers in FreeHand, and select Layers in the FreeHand Import dialog box in Flash. (If overlapping objects on a single layer are imported into Flash, the overlapping shapes are divided at intersection points, just as with overlapping objects that you create in Flash.)
- Flash can support up to eight colors in a gradient fill. If a FreeHand file contains a gradient fill with more than eight colors, Flash creates clipping paths to simulate the appearance of a gradient fill. Clipping paths can increase file size. To minimize file size, use gradient fills with eight colors or fewer in FreeHand.
- Flash imports each step in a blend as a separate path. Thus, the more steps a blend has in a FreeHand file, the larger the imported file size is in Flash.
- When you import files with strokes that have square caps, Flash converts the caps to round caps.
- Flash converts placed grayscale images to RGB images. This conversion can increase the imported file’s size.
- When you import files with placed EPS images, select the Convert Editable EPS When Imported option in FreeHand Import Preferences before you place the EPS into FreeHand. If you do not select this option, the EPS image is not viewable when imported into Flash. In addition, Flash does not display information for an imported EPS image (regardless of the Preferences settings used in FreeHand).

1 Select File > Import > Import To Stage or File > Import To Library.

2 Select FreeHand from the Files Of Type (Windows) or Show (Macintosh) pop-up menu.

3 Navigate to a FreeHand file and select it.

4 Click Open.

5 For Mapping Pages, select a setting:

Scenes Converts each page in the FreeHand document to a scene in the Flash document.

Keyframes Converts each page in the FreeHand document to a keyframe in the Flash document.

6 For Mapping Layers, select one of the following:

Layers Converts each layer in the FreeHand document to a layer in the Flash document.

Keyframes Converts each layer in the FreeHand document to a keyframe in the Flash document.

Flatten Converts all layers in the FreeHand document to a single flattened layer in the Flash document.

7 For Pages, do one of the following:

- To import all pages from the FreeHand document, select All.
- To import a page range from the FreeHand document, enter page numbers for From and To.

8 For Options, select any of the following options:

Include Invisible Layers Imports all layers (visible and hidden) from the FreeHand document.

Include Background Layer Imports the background layer with the FreeHand document.

Maintain Text Blocks Preserves text in the FreeHand document as editable text in the Flash document.

9 Click OK.

About AutoCAD DXF files

Flash supports the AutoCAD DXF format in AutoCAD 10.

DXF files do not support the standard system fonts. Flash tries to map fonts appropriately, but the results can be unpredictable, particularly for text alignment.

Because the DXF format does not support solid fills, filled areas are exported as outlines only. For this reason, the DXF format is most appropriate for line drawings, such as floor plans and maps.

You can import two-dimensional DXF files into Flash. Flash does not support three-dimensional DXF files.

Although Flash doesn't support scaling in a DXF file, all imported DXF files produce 12-inch x 12-inch files that you can scale using Modify > Transform > Scale. Also, Flash supports only ASCII DXF files. If your DXF files are binary, convert them to ASCII before importing them into Flash.

Working with Illustrator AI files

About Adobe Illustrator AI files

Flash lets you import Illustrator AI files, and to a large extent preserves your artwork's editability and visual fidelity. The AI Importer also provides you with a great degree of control in determining how your Illustrator artwork is imported into Flash, letting you specify how to import specific objects into an AI file.

The Flash AI Importer provides the following key features:

- Preserves editability of the most commonly used Illustrator effects as Flash filters.
- Preserves editability of blend modes that Flash and Illustrator have in common.
- Preserves the fidelity and editability of gradient fills.
- Maintains the appearance of RGB (red, green, blue) colors.
- Imports Illustrator Symbols as Flash Symbols.
- Preserves the number and position of Bezier control points.
- Preserves the fidelity of clip masks.
- Preserves the fidelity of pattern strokes and fills.
- Preserves object transparency.

- Converts the AI file layers to individual Flash layers, keyframes, or a single Flash layer. You can also import the AI file as a single bitmap image, in which case Flash flattens (rasterizes) the file.
- Provides an improved copy-and-paste workflow between Illustrator and Flash. A copy-and-paste dialog box provides settings to apply to AI files being pasted onto the Flash stage.

For video tutorials, see:

- Using symbols between Illustrator and Flash: www.adobe.com/go/vid0198
- Using text between Illustrator and Flash: www.adobe.com/go/vid0199

Compatibility between Flash and Illustrator

Certain visual attributes can either not be accurately imported, or, after they are imported, lose their ability to be further edited in the Flash authoring environment. The AI Importer provides you with several options to import and place artwork to best maintain its visual appearance and editability. However, certain visual attributes cannot be preserved. Use the following guidelines to improve the appearance of AI files imported into Flash:

- Flash supports only the RGB color space, and not the CMYK color space, which is common in printing. Flash can convert CMYK images to RGB, however, colors are better preserved if you convert the colors to RGB in Illustrator.
- To preserve the drop shadow, inner glow, outer glow, and Gaussian blur AI effects as editable Flash filters, Flash imports the objects to which they are applied as a Flash movie clip. If you attempt to import an object with these attributes as something other than a movie clip, Flash displays an incompatibility alert, and recommends importing the object as a movie clip.

Import Adobe Illustrator files

Flash can import Illustrator AI files in version 10 or earlier. If the raster file in Illustrator is linked, only JPEG, GIF, or PNG is imported with native format reserved. All other files are converted to PNG format in Flash. Additionally, conversion to PNG depends on the version of QuickTime installed.

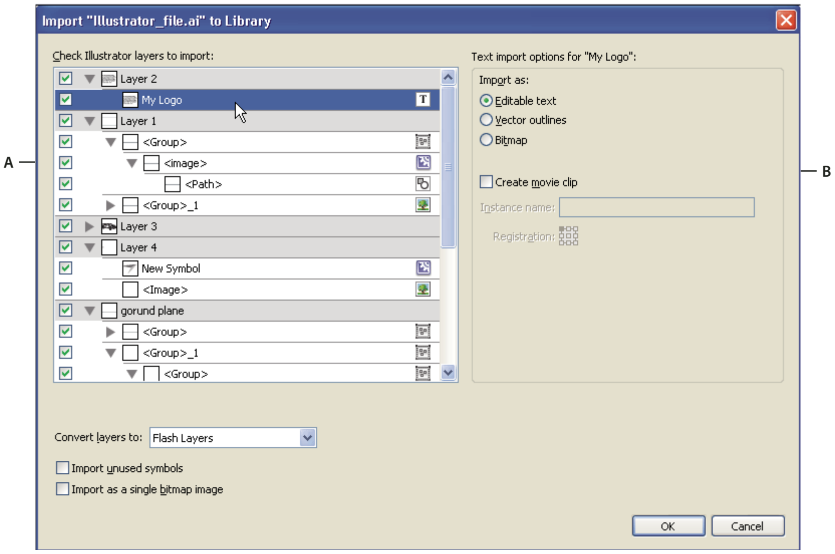
***Note:** To import an Illustrator EPS or Adobe Acrobat PDF file, open the file in Illustrator CS 3, save it as a CS 3-compatible AI file, and import the resulting AI file into Flash*

***Note:** The AI Importer was developed to import AI files created with Illustrator CS3. While there are no known issues importing AI files created in earlier versions of Illustrator, we recommend importing AI files created using Illustrator CS 3. If you encounter issues importing AI files created in earlier version of Illustrator, open the file in Illustrator CS 3, and save the AI file as a CS 3 compatible file and re-import it into Flash.*

For video tutorials, see:

- Importing Illustrator files: www.adobe.com/go/vid0197
- Using symbols between Illustrator and Flash: www.adobe.com/go/vid0198
- Using text between Illustrator and Flash: www.adobe.com/go/vid0199

In certain instances, you may need to ungroup all the Illustrator objects on all layers when Flash cannot group items on import. After all the objects are ungrouped, they can be manipulated like any other Flash object.


A
B
Import "Illustrator_file.ai" to Library
Check Illustrator layers to import:
OK
Cancel

☑ ▼ Layer 2
☑ My Logo
☑ ▼ Layer 1
☑ ▼ <Group>
☑ ▼ <image>
☑ <Path>
☑ ▶ <Group>_1
☑ ▶ Layer 3
☑ ▼ Layer 4
☑ New Symbol
☑ <Image>
☑ ▼ gorund plane
☑ ▶ <Group>
☑ ▼ <Group>_1
☑ ▼ <Group>

Text import options for "My Logo":

Import as:

● Editable text
○ Vector outlines
○ Bitmap

□ Create movie clip

Instance name:

Registration:

Convert layers to: Flash Layers

□ Import unused symbols
□ Import as a single bitmap image

Illustrator AI Importer dialog box
A. *Layers panel* ***B.*** *Import options available for the selected object*

Choose from the following options when importing Adobe Illustrator files:

- Convert layers to Flash layers or keyframes, or single Flash layers.
- Import text as editable text, vector outlines, or bitmaps.
- Import as a single bitmap image. Choosing this option imports the AI file as a single bitmap image, and disables the layers list and import options within the AI import dialog box.

1 Select File > Import To Stage or Import To Library.

2 Navigate to the AI file to import, select it, and click OK. The Import Illustrator Document To Stage or The Import Illustrator Document To Library dialog box appears.

This dialog box provides options to import the Illustrator file. Depending on the types of objects in the Illustrator file you're importing, the options available to you vary.

3 (Optional) To generate a list of items in the AI file that are incompatible with Flash, click Incompatibility Report. The Incompatibility Report button appears only if incompatibilities exist with Flash in the AI file.

The Incompatibility Report analyzes possible incompatibilities between Illustrator and Flash. The Import Options area (next to the Alert button) of the AI Importer dialog box displays recommendations to obtain the maximum compatibility for any incompatible item.

The Incompatibility Report includes an Apply Recommended Import Settings checkbox. If checked, Flash automatically applies the recommended import options to any incompatible objects within the AI file. The exceptions to this, are when the AI document is larger in size than that supported by Flash, and when the AI document uses the CMYK color mode. To correct either of these incompatibilities, reopen the document in Illustrator CS 3, and adjust the size of the document, or change the color mode to RGB.

4 (Optional) In the AI Import dialog box, select layers, groups, and individual objects, and choose how to import each item. For information on the import options available for the different types of Illustrator layers and objects, see "Illustrator object import options" on page 145.

5 For Convert Layers To, select one of the following:

Flash Layers Converts each layer in the imported document to a layer in the Flash document.

Keyframes Converts each layer in the imported document to a keyframe in the Flash document.

Single Flash Layer Converts all layers in the imported document to a single flattened layer in the Flash document.

6 For the remaining options, select from the following:

Place Objects At Original Position The contents of the AI file retain the exact position that they had in Illustrator. For example, if an object was positioned at X = 100 Y = 50 in Illustrator, it assumes the same coordinates on the Flash Stage.

If this option is not selected, the imported Illustrator layers are centered within the current view. The items in the AI file remain relative to each other when imported; however, all objects are centered as one block in the current view. This feature can be useful if you are zoomed in to an area of the Stage, and are importing a specific object for that area of the Stage. If you imported the object using the original coordinates, you might not see it imported, because it might be placed outside of the current stage view.

***Note:** When importing the AI file into the Flash library, this option is not available.*

Set Stage Size To Same Size As Illustrator Artboard The Flash stage size resizes to the same size as the Illustrator artboard (or active crop area) used to create the AI file. This option is not selected by default.

***Note:** When importing the AI file into the Flash library, this option is not available.*

Import Unused Symbols Any symbols in the AI file's library that have no instances on the artboard are imported into the Flash library. If this option is not selected, the unused symbols are not imported into Flash.

Import As A Single Bitmap Image Imports the AI file as a single bitmap image, and disables the layers list and import options within the AI import dialog box.

7 Click OK.

See also

"Import Photoshop PSD files" on page 149

"Exporting Flash content, images, and video" on page 447

Importing AI files to the Flash library

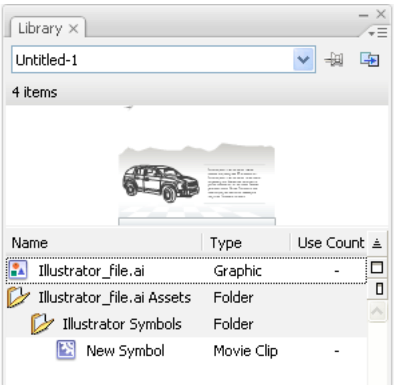
Importing an AI file into the library is similar to importing to the Stage, the difference being that the entire AI file is encapsulated as a Flash symbol. The content is imported into the library and organized by the layering and grouping structure of the AI file.

When you import an AI file into the library, the root folder uses the name of the AI file. After the AI file is imported into the library, you can change the name of the root folder, or move the layers out of the folder.

***Note:** The Library orders the contents of the imported AI file alphabetically. The hierarchical grouping and folder structure remains the same, but the library reorders them alphabetically.*

When converting AI layers to keyframes, the AI file is imported as a movie clip; converting the AI layers to Flash layers, or as a single Flash layer, imports the AI file as a graphic symbol. The resulting movie clip or graphic symbol contains all of the content of the AI file imported to its timeline, as if the content were imported to the stage. Almost all movie clips have a bitmap or other asset associated with them. To minimize confusion and naming conflicts, these assets are stored in an Assets folder in the same folder as the movie clip.

Note: When you import to the library, the AI file's contents are imported to the movie clip's timeline, not the main Flash timeline.



4 items

Name	Type	Use Count
Illustrator_file.ai	Graphic	-
Illustrator_file.ai Assets	Folder	
Illustrator Symbols	Folder	
New Symbol	Movie Clip	-

Flash Library after importing an AI file.

AI File Importer preferences

The Flash Preferences dialog box lets you set import preferences for AI files and the AI File Importer dialog box. The preferences you specifiy for importing AI files affects the options the AI Import dialog box is initially populated with for the Illustrator object types.

Note: To override the preferences specified for different layer types on an object by object basis, use the AI Import dialog box. Select the layer, object, or group to change its import options, and specify the necessary options.

General AI File Importer Preferences Preferences that affect how the AI Importer will respond when importing AI files:

- **Show Import Dialog Box** Specifies that the AI File Importer dialog box appears.

- **Exclude Object Outside Crop Area** Excludes objects on the Illustrator canvas which fall outside the artboard or crop area.

- **Import Hidden Layers** Specifies that hidden layers be imported by default.

Import Text As Lets you specify the following import preferences for text objects:

- **Editable Text** Specifies that Illustrator text import as editable Flash text. The appearance of the text may be compromised to maintain the editability of the text.

- **Vector Outlines** Converts text to vector paths. Use this option to preserve the visual appearance of the text. Some visual effects might be sacrificed—such as unsupported blend modes and filters—but visual attributes such as text on a path are maintained if the text is imported as a movie clip. The text itself is no longer editable, but opacity and compatible blend modes maintain their editability.

Note: To preserve the drop shadow, inner glow, outer glow, and Gaussian blur that AI effects applied to text as editable Flash filters, select Create Movie Clip Import to import the text as a movie clip.

- **Bitmaps** Rasterizes the text into a bitmap to preserve the exact appearance of the text as it was in Illustrator. If filters or other effects are applied that are not compatible with Flash, importing the text as a bitmap preserves the visual appearance. Rasterized text is no longer editable.

- **Create Movie Clips** Specifies that text objects be imported inside a movie clip. To maintain supported blend modes, AI effects, and transparency of less than 100% between Illustrator and Flash, specify that the text object be imported as a movie clip.

Import Paths As Lets you specify the following path import preferences:

- **Editable Paths** Creates an editable vector path. Supported blend modes, effects, and object transparency are preserved, but attributes not supported in Flash are discarded.

- **Bitmaps** Rasterizes the path into a bitmap to preserve the exact appearance of the path in Illustrator. A rasterized image is no longer editable.

- **Create Movie Clips** Specifies that path objects be imported inside a movie clip.

Images Lets you specify the import preference for images:

- **Flatten Bitmaps To Maintain Appearance** Rasterizes the image into a bitmap to preserve the appearance of blending modes and effects that are not supported in Flash. A rasterized image is no longer editable.

- **Create Movie Clips** Specifies that images be imported inside a movie clip.

Groups Lets you specify the import preferences for groups:

- **Import As Bitmaps** Rasterizes the group into a bitmap to preserve the appearance of the objects as they appeared in Illustrator. After a group is converted to a bitmap, the objects in it cannot be selected or renamed.

- **Create Movie Clips** Specifies that all objects in the group be encapsulated into a single movie clip.

Layers Lets you specify the import preferences for layers:

- **Import As Bitmaps** Rasterizes the layer into a bitmap to preserve the appearance of the objects as they appeared in Illustrator.

- **Create Movie Clips** Specifies that the layer is encapsulated into a movie clip.

Movie Clip Registration Specifies a global registration point for movies that are created. This setting applies to the registration point for all object types. This option can be changed on an object by object basis in the AI File Importer dialog box; this is the initial setting for all object types. For more information on movie clip registration, see “Edit symbols” on page 210.

See also

“Set preferences in Flash” on page 27

Copy and pasting between Illustrator and Flash

If you copy-and-paste (or drag-and-drop) artwork between Illustrator and Flash, the Paste dialog box appears, which provides import settings for the AI file being copied (or pasted).

Paste As Bitmap Flattens the file being copied into a single bitmap object.

Paste Using AI File Importer Preferences Imports the file using the AI file import setting specified in Flash Preferences (Edit > Preferences).

Apply Recommended Import Settings To Resolve Incompatibilities Enabled by default when Paste Using AI File Importer Preferences is selected. Automatically fixes any incompatibilities detected in the AI file.

Maintain Layers Enabled by default when Paste Using AI File Importer Preferences is selected. Specifies that layers in the AI file be converted to Flash layers (the same as if you selected Convert To Flash Layers from the AI Import dialog box). If deselected, all layers are flattened into a single layer.

Illustrator object import options

Layers manage all the items that make up Illustrator artwork. By default, all items are organized in a single, parent layer. Import all items in a given parent layer as a single, flattened bitmap, or, individually select each object and specify import options specific to the type of object it is (text, path, group, and so on). The AI Importer provides options to select layers in the artwork you are importing, and specify individual import options based on preserving the visual appearance of an object, or its editability in Flash.

Correcting incompatible graphic effects

1 To generate an incompatibility report, click Incompatibility Report. The incompatibility report lists items in the AI file that are incompatible with Flash.

2 Select Change Import Settings To Resolve Object Incompatibilities. Many incompatibilities between Illustrator and Flash can be automatically corrected using the incompatibility report, and the import recommendations suggested in the Import options area of the AI Import dialog box.

Select individual objects

1 Select the object to specify import options for. Illustrator objects that you can select include layers, groups, individual paths, text, and images.

2 Review the import options available for the type of object you've selected in the object options section of the dialog box. Note if any incompatibilities are listed, and what the recommended correction is to import the object.

3 Select the desired import options, and select another object to specify import options for, or click OK.

Selecting layers

The Layers panel lists the objects in a document. By default, every Illustrator document contains at least one layer, and each object in a given file is listed under that layer.

When an item in the Layers panel contains other items, a triangle appears to the left of the item's name. To show or hide the content, click the triangle. If no triangle appears, the layer contains no additional items.

Note: *You can expand or collapse all groups and layers using the AI Importer context menu. Right-click to display the context menu, and select Expand All or Collapse All.*

Selection column Controls whether items are selected for import or not. If an item is checked, you can select that layer and specify import options; if Edit is not checked, the layer is dimmed, and you cannot specify any import setting for the item in that layer.

Object Type column An icon indicates what the Flash object type located on that layer will be when imported, and, if visible, that the item is selected. The object types are:

- Text T
- Path
- Group
- Movie clip
- Graphic symbol
- Image

Text import options

Flash lets you import text as editable text, vector outlines, and a flattened bitmap. To maintain supported blend modes, AI effects, and transparency of less than 100% between Illustrator and Flash, use import text as a movie clip. Importing text as a movie clip preserves the editability of compatible visual effects.

Editable Text By default, Illustrator text imports as editable Flash text. The appearance of the text can be compromised to maintain the editability of the text.

Vector Outlines Converts text to vector paths. Use this option to preserve the visual appearance of the text. Some visual effects might be sacrificed—such as unsupported blend modes and filters—but visual attributes such as text on a path are maintained if the text is imported as a movie clip. The text itself is no longer editable, but opacity and compatible blend modes maintain their editability.

***Note:** To preserve the drop shadow, inner glow, outer glow, and Gaussian blur that AI effects applied to text as editable Flash filters, select Create Movie Clip Import to import the text as a movie clip.*

Bitmap Rasterizes the text into a bitmap to preserve the exact appearance of the text as it was in Illustrator. If filters or other effects are applied that are not compatible with Flash, importing the text as a bitmap preserves the visual appearance. Rasterized text is no longer editable.

Path import options

A path is the line that results from drawing in Illustrator. A path is either open, like an arc, or closed, like a circle. For an open path, the starting and ending anchor points for the path are called endpoints. Editable paths can be imported into Flash, however, if certain blend modes, filters, or other effects are applied to the path, those effects might not be compatible with Flash.

Bitmap Rasterizes the path into a bitmap to preserve the exact appearance of the path in Illustrator. A rasterized image is no longer editable.

Editable Path Creates an editable vector path. Supported blend modes, effects, and object transparency are preserved, but attributes not supported in Flash are discarded.

Image import options

Bitmap images are the most common electronic medium for continuous-tone images, such as photographs or digital paintings. Illustrator creates bitmap effects by using filters, effects, and graphic styles. While many of these effects are compatible with Flash, some might need to be flattened—or rasterized—to preserve their visual appearance.

***Note:** If a raster file in Illustrator is linked, only JPEG, GIF, or PNG is imported with their native format preserved. All other file types are converted to the PNG format in Flash. Additionally, the conversion (to PNG) depends on the version of QuickTime installed on your computer.*

Flatten Bitmap To Maintain Appearance Rasterizes the image into a bitmap to preserve the appearance of blending modes and effects that are not supported in Flash. A rasterized image is no longer editable.

Create Movie Clip Imports the Illustrator images as movie clips.

Group import options

Groups are collections of graphic objects that are treated as a single unit. Grouping lets you move or transform a number of objects without affecting their attributes or relative positions. For example, you might group the objects in a logo design so that you can move and scale the logo as one unit. Groups can also be nested. That is, they can be grouped in other objects or groups to form larger groups.

Groups appear as <Group> items in the Import panel. When an item such as a group contains other items, a triangle appears to the left of the item's name. Click the triangle to show or hide the contents of the group. If no triangle appears, the item contains no additional items.

Import As Bitmap Rasterizes the group into a bitmap to preserve the appearance of the objects as they appeared in Illustrator. After a group is converted to a bitmap, the objects in it cannot be selected or renamed.

Create Movie Clip Encapsulates all objects in the group into a single movie clip.

Working with Photoshop PSD files

Working with Photoshop and Flash

You can use Adobe® Photoshop® and Adobe® Flash® together to create visually compelling web-based applications, animations, or interactive messaging elements. Photoshop lets you create still images and artwork, providing a high degree of creative control. Flash lets you bring those still images together and incorporate them into interactive Internet content.

The Photoshop drawing and selection tools provide you with a greater degree of creative control than the tools found in Flash. If you have to create complex visual images, or retouch photographs for use in interactive presentations, use Photoshop to create your artwork, and then import the finished images into Flash.

Importing still images into Flash

Flash can import still images in many formats, but you usually use the native Photoshop PSD format when importing still images from Photoshop into Flash.

When importing a PSD file, Flash can preserve many of the attributes that were applied in Photoshop, and provides options for maintaining the visual fidelity of the image and further modifying the image. When you import a PSD file into Flash, you can choose whether to represent each Photoshop layer as a Flash layer, individual keyframes, or a single, flattened image. You can also encapsulate the PSD file as a movie clip.

Exchanging movies

You can exchange QuickTime video files between Photoshop and Flash. For example, you can render a QuickTime movie directly from Photoshop, and then import it into Flash, converting it into an FLV (Flash Video) file, which can be played in Flash® Player.

When you use Photoshop to modify video footage, you can paint nondestructively on its frames. When you save a Photoshop file with a video layer, you are saving the edits that you made to the video layer, not edits to the footage itself.

***Note:** When you import a QuickTime video file from Photoshop to Flash, use the Import Video dialog box (File > Import Video). Using the Photoshop PSD import feature to import video only imports the first frame of a video file.*

You can also export Flash documents as QuickTime video, and import them into Photoshop where you can paint nondestructively on the video frames. For example, you can create an animated sequence in Flash, export the Flash document as a QuickTime video, and then import the video into Photoshop.

Color

Flash works internally with colors in the RGB or HSB (hue, saturation, brightness) color space. Although Flash can convert CMYK images to RGB, you should create Photoshop artwork in RGB. Before you import CMYK artwork from Photoshop into Flash, convert the image to RGB in Photoshop.

See also

"Import Photoshop PSD files" on page 149

"Exporting QuickTime" on page 453

About Photoshop PSD files

Flash lets you import Photoshop PSD files, and preserves most of your artwork's data. The PSD Importer also lets you control how your Photoshop artwork is imported into Flash, both letting you specify how to import specific objects in a PSD file, as well as specifying that the PSD file be converted to a Flash movie clip.

The Flash PSD Importer provides the following key features:

- PSD files imported into Flash maintain their color fidelity from Photoshop.
- Preserves editability of blend modes that both Flash and Photoshop have in common.
- Smart Objects in the PSD file are rasterized, and imported into Flash as bitmaps, which preserves object transparency.
- Converts the PSD file layers to individual Flash layers or keyframes. Or, import the PSD file as a single bitmap image, in which case Flash flattens (rasterizes) the file.
- Drag-and-drop from Photoshop to Flash invokes the PSD file importer, letting you choose how to import your Photoshop artwork.

For a video tutorial about designing website with Photoshop and Flash, see www.adobe.com/go/vid0201.

Compatibility between Flash and Photoshop

Certain visual attributes can either not be accurately imported, or, after they are imported, lose their ability to be further edited in the Flash authoring environment. The PSD Importer provides you with several options to import and place artwork to best maintain its visual appearance and editability. However, certain visual attributes cannot be preserved. Use the following guidelines to improve the appearance of PSD files imported into Flash:

- Flash supports only the RGB color space, and not the CMYK color space, which is common in printing. Flash can convert CMYK images to RGB, however, colors are better preserved if you convert the colors to RGB in Photoshop.
- Flash can import the following photoshop blend modes, and maintain their editability: Normal, Darken, Multiply, Lighten, Screen, Hard Light, Difference, and Overlay.

If you use a blend mode that Flash does not support, you can rasterize the layer to maintain its visual appearance, or remove the blend mode from the layer.

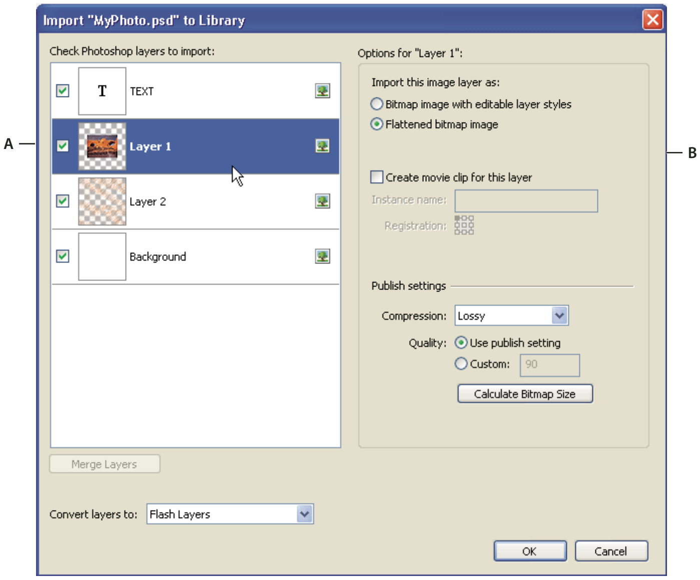
- Flash cannot import Photoshop Smart Objects as editable objects. To preserve the visual attributes of Smart Objects, they are rasterized and imported into Flash as bitmaps.
- Flash can only import the first frame of Photoshop Video layers.
- Image and Fill layers are always rasterized when imported into Flash.
- When importing objects containing transparent areas as flattened bitmaps, any objects on the layers behind the transparent portion of the object will be visible through the transparent area—assuming the objects behind the object containing transparency are also being imported. To prevent this, import only the transparent object as a flattened bitmap.

To import multiple layers and maintain transparency without having any visible remnants of the layers behind the transparency, import the PSD file using the Bitmap Image With Editable Layer Styles option. This will encapsulate the imported objects as a movie clip, and use the movie clip's transparency. This is particularly useful if you need to animate the different layers in Flash.

Import Photoshop PSD files

Photoshop format (PSD) is the default Photoshop file format. Flash can directly import PSD files and preserve many Photoshop features, retaining the image quality and editability of the PSD file in Flash. You can also flatten PSD files when importing them, creating a single bitmap image file that retains the image's visual effects, but removes the hierarchical layer information native to the PSD file format.

For a video tutorial about designing website with Photoshop and Flash, see www.adobe.com/go/vid0201.



Import "MyPhoto.psd" to Library

Check Photoshop layers to import:

☑ T TEXT

☑ Layer 1

☑ Layer 2

☑ Background

Merge Layers

Convert layers to: Flash Layers

Options for "Layer 1":

Import this image layer as:

○ Bitmap image with editable layer styles

● Flattened bitmap image

□ Create movie clip for this layer

Instance name:

Registration:

Publish settings

Compression: Lossy

Quality: ● Use publish setting

○ Custom: 90

Calculate Bitmap Size

OK Cancel

The PSD importer dialog box
A. *Layers in the PSD file being imported* ***B.*** *Import options available for the selected layer or object.*

1 Select File > Import To Stage or Import To Library.

2 Navigate to the Adobe Photoshop PSD file to import, select it, and click OK.

3 (Optional) In the PSD Import dialog box, select layers, groups, and individual objects and choose how to import each item.

4 For Convert Layers To, select one of the following:

Flash Layers All selected layers in the Select Photoshop layers list are placed on their own layer. Each layer is labeled with the name of the layer in the Photoshop file. The layers in Photoshop are objects on the individual layers. The objects also have the name of the layer in Photoshop when put into the Library.

Keyframes All selected layers in the Select Photoshop layers list are placed in individual keyframes on a new layer. The new layer is named for the Photoshop file (for example, myfile.psd). The layers in Photoshop become objects on the individual keyframes. The objects also have the name of the layer in Photoshop when put into the Library.

5 For the remaining options, select from the following:

Place Layers At Original Position The contents of the PSD file retain the exact position that they had in Photoshop. For example, if an object was positioned at X = 100 Y = 50 in Photoshop, it assumes the same coordinates on the Flash stage.

If this option is not selected, the imported Photoshop layers are centered on the stage. The items in the PSD file remain relative to each other when imported; however, all objects are centered as one block in the current view. This feature can be useful if you are zoomed in to an area of the Stage, and are importing a specific object for that area of the stage. If you imported the object using the original coordinates, you might not see it imported, because it might be placed outside of the current stage view.

Note: *When you import the PSD file to the Flash library, this option is not available.*

Set Stage Size To Same Size As Photoshop Canvas The Flash Stage size resizes to the same size as the Photoshop document size (or active crop area) used to create the PSD file. This option is not selected by default.

Note: *When you import the PSD file to the Flash library, this option is not available.*

6 Click OK.

Importing PSD files to the Flash library

Importing a PSD file into the library is similar to importing to the Stage. When you import a PSD file into the library, the root folder uses the name of the PSD file. After the PSD file is imported into the library, you can change the name of the root folder, or move the layers out of the folder.

Note: *The library orders the contents of the imported PSD file alphabetically. The hierarchical grouping and folder structure remains the same, but the library reorders them alphabetically.*

A movie clip is created that contains all of the content of the PSD file imported to its timeline, as if the content were imported to the stage. Almost all movie clips have a bitmap or other asset associated with them. To minimize confusion and naming conflicts, these assets are stored in an Assets folder in the same folder as the movie clip.

Note: *When you import to the library, the PSD file's contents are imported to the movie clip's timeline, not the main Flash timeline.*

PSD file import preferences

The Flash Preferences dialog box lets you set import preferences for PSD files. The preferences you specify for importing PSD files affects the options the PSD Import dialog box is initially populated with for the Photoshop layer types.

Note: *To override the preferences specified for different layer types on an object by object basis, use the PSD Import dialog box. Select the layer to change import options for, and specify the necessary options.*

See also

“Set preferences in Flash” on page 27

Image Layer Import Preferences

These options specify how the import options for image layers will initially be set.

Bitmap Images With Editable Layer Styles Creates a movie clip with a bitmap clipped inside. Specifying this option maintains supported Blend modes and opacity, but other visual attributes that cannot be reproduced in Flash will be removed. If this option is selected, the object must be converted into a movie clip.

Flattened Bitmap Images Rasterizes the text into a flattened bitmap image to maintain the exact appearance the text layer had in Photoshop.

Create Movie Clips Specifies that the image layers be converted to a movie clip when imported into Flash. This option can be changed in the PSD Import dialog box on an object by object basis if you do not want all of the image layers to be movie clips.

Text layer import preferences

These options specify how the import options for image layers will initially be set.

Editable Text Creates an editable text object from the text on the Photoshop text layer. The appearance of the text will be sacrificed to maintain the editability of the text. If this option is selected, the object must be converted into a movie clip.

Vector Outlines Vectorizes the text into paths. The appearance of the text may be altered, but visual attributes will be maintained. If this option is selected, the object must be converted into a movie clip.

Flattened Bitmap Images Rasterize the text to maintain the exact appearance the text layer had in Photoshop.

Create Movie Clips Automatically converts the text layer to a movie clip when imported into Flash. This option can be changed in the PSD Import dialog box on an object by object basis if you do not want all of the text layers to be movie clips. This option is enabled by default when any option other than Flattened Bitmap Images is selected, as the other options have to be converted to a movie clip.

Text layer import preferences

These options specify how the import options for text layers will initially be set.

Editable Text Creates an editable text object from the text on the Photoshop text layer. The appearance of the text will be sacrificed to maintain the editability of the text. If this option is selected, the object must be converted into a movie clip.

Vector Outlines Vectorizes the text into paths. The appearance of the text may be altered, but visual attributes will be maintained. If this option is selected, the object must be converted into a movie clip.

Flattened Bitmap Images Rasterize the text to maintain the exact appearance the text layer had in Photoshop.

Create Movie Clips Automatically converts the text layer to a movie clip when imported into Flash. This option can be changed in the PSD Import dialog box on an object by object basis if you do not want all of the text layers to be movie clips. This option is enabled by default when any option other than Flattened bitmap images is selected, as the other options have to be converted to a movie clip.

Shape layer import preferences

These options specify how the import options for shape layers will initially be set.

Maintain Editable Paths And Layer Styles This option will create an editable vector shape with a bitmap clipped inside the vector shape. Supported blend modes and opacity will also be maintained with this option, but other visual attributes that cannot be reproduced in Flash will be sacrificed. If this option is selected, then the object must be converted into a movie clip.

Convert To Bitmap This option will rasterize the shape and maintain the exact appearance the shape layer had in Photoshop.

Create Movie Clips This option will set the shape layers to be converted to a movie clip when imported into Flash. This option can be changed on an object by object basis, if you do not want some shape layers to be movie clips. This option is disabled if the Maintain editable paths and layers styles checkbox is checked.

Layer group import preferences

This option specifies how the options for layer groups will initially be set.

Create Movie Clips Specifies that all groups be converted to a movie clip when imported into Flash. This can be changed on an object by object basis if you do not want some layer groups to be movie clips.

Merged bitmap import preferences

This option specifies how the import options for merged bitmaps will initially be set.

Create Movie Clips This option will set the shape layers to be converted to a movie clip when imported into Flash. This option can be changed on an object by object basis, if you do not want some merged bitmaps to be movie clips. This option is disabled if the Maintain Editable Paths And Layers Styles checkbox is checked.

Movie clip registration import preferences

Specifies a global registration point for movies that are created. This setting applies to the registration point for all object types. This option can be changed on an object by object basis in the PSD Import dialog box; this is the initial setting for all object types. For more information on movie clip registration, see "Edit symbols" on page 210.

Publish setting import preferences

The Publish settings preferences let you specify the degree of compression and document quality to apply to the image when publishing the Flash document as a SWF file. These settings take effect only when you publish the document as a SWF file, and have no effect on the image when you import it to the Flash stage or library.

Compression Lets you choose either lossy or lossless compression formats:

- **Lossy** Lossy (JPEG) compresses the image in JPEG format. To use the default compression quality specified for the imported image, select Use Publish Setting. To specify a new quality compression setting, select the Custom option, and enter a value between 1 and 100 in the Quality text field. (A higher setting preserves greater image integrity but yields a larger file size.).
- **Lossless** Lossless (PNG/GIF) compresses the image with lossless compression, in which no data is discarded from the image.

***Note:** Use Lossy compression for images with complex color or tonal variations, such as photographs or images with gradient fills. Use Lossless compression for images with simple shapes and relatively few colors.*

Photoshop Import options

When you import a Photoshop PSD file that contains multiple layers, you can set the following options:

Layer Comp If the Photoshop file contains layer comps, you can specify which version of the image to import. A *layer comp* is a snapshot of a state of the Photoshop Layers palette. Layer comps record three types of layer options, all of which are imported into Flash:

- **Layer visibility**: whether a layer is displayed or hidden.
- **Layer position** in the document.
- **Layer appearance**: whether a layer style is applied to the layer and the layer's blending mode.

If no layer comps are present, this pop-up menu is hidden. Flash supports all aspects of the layer comp's fidelity, including visibility, position, and layer style.

Select Photoshop Layers Lists all layers, groups, and layer effects in an image. To select which layers to import, use the options to the left of the layer's thumbnail. By default, all layers visible in Photoshop are checked, and invisible layers are not checked.

***Note:** Adjustment layers do not have a compatible layer type in Flash. For this reason, when importing as a flattened bitmap, the visual effect of the adjustment layer is applied to maintain the appearance of the Photoshop layer being imported. If you've selected other import options, the adjustment layers will not be applied.*

Merge Layers Merge (or collapse) two or more layers into a single bitmap, and then import the resulting single bitmap object rather than individual objects. You can only merge layers that are on the same level, and the selection must be contiguous. For example, you cannot select an item inside a folder and an item outside a folder and merge them. Instead, select the folder and the item outside the folder to merge them. You cannot Control-click individual items that are out of sequence and merge them.

***Note:** If you select a merged bitmap object, the Merge Layers button changes to the Separate button. To separate any merged bitmap objects you create, select the resulting single bitmap, and click the Separate button.*

Importing text objects

Text objects are text layers in PhotoShop. Choose how to import text into Flash.

Editable Text Creates an editable text object from the text on the selected Photoshop layer. The appearance of the text might be compromised to maintain the editability of the text. If you import the text as a movie clip, the movie clip contains an editable text object.

***Note:** When importing editable text into the library, it must be inside a movie clip. Only movie clips, bitmaps, and graphic symbols can be stored in the library. When Editable Text is selected as an option for a text layer being imported into the library, it is automatically imported as a graphic symbol.*

Vector Outlines Converts text to vector paths to preserve the visual appearance of the text. The text itself is no longer editable, but opacity and compatible blend mode maintain their editability. If this option is selected, the object must be converted to a movie clip.

Flattened Bitmap Image Rasterizes the text into a bitmap to preserve the exact appearance of the text layer in Photoshop. Rasterized text is no longer editable.

***Note:** When importing text on a path you must import it as a flattened bitmap image to preserve the visual fidelity of the object.*

Importing shape objects

A Shape Layer object is an object that was originally a shape layer in Photoshop or an image layer with a vector clipping mask on it.

Editable Paths And Layer Styles Creates an editable vector shape with a bitmap clipped inside the vector. Supported blend modes, filters, and opacity are maintained. Unsupported blend modes that cannot be reproduced in Flash are removed. The object must be converted to a movie clip.

Flattened Bitmap Image Rasterizes the shape into a bitmap to preserve the exact appearance of the shape layer in Photoshop. A rasterized image is no longer editable.

Importing image or fill layers

If the image or fill layer is associated with a vector mask, it is treated as a shape layer object.

Bitmap Image With Editable Layer Styles Creates a movie clip with a bitmap inside. Supported Blend modes, filters, and opacity are maintained. Unsupported blend modes that cannot be reproduced in Flash are removed. The object must be converted to a movie clip.

Flattened Bitmap Image Rasterizes the image into a bitmap to preserve the exact appearance of the image or fill layer in Photoshop.

Importing merged bitmap objects and objects in a merged bitmap

A *merged bitmap* is an object that contains more than one Photoshop layer that is flattened (or merged) into a single bitmap when imported into Flash. Objects in a merged bitmap represent layers in Photoshop. To create a merged bitmap, select two or more layers, and select the Merge Layers button.

Importing multiple objects of different types

If you import multiple objects of different types, Flash only lets you import the selected objects with the import options they share in common, such as Create Movie Clip and Registration.

Importing multiple objects of the same type

If you import multiple objects of the same type, the import options displayed are the same as if a single object of that type was selected. If the objects do not share the same attributes, the import options displayed are in an indeterminate state, and your results may not be as expected.

Importing a Group folder

When you import a group folder, you can import it as a movie clip, or place each layer in the group on its own layer or keyframe on the timeline.

If you select Import As Movie Clip, each layer in the group folder is placed on a layer in a movie clip, which is then placed on its own layer or keyframe on the timeline. The movie clip uses the same name as that group folder had in Photoshop, and if you import the movie clip to a Flash layer, the layer uses the same name.

If you do not place the group in a movie clip, each layer is converted to the type that is currently set for it, and each layer in the group is imported to its own Flash layer. The Flash layers are named for the individual layers in the PSD file.

Importing and merging layers

The PSD importer lets you merge two or more layers into a merged bitmap that is imported as a single bitmap file rather than individual objects.

The layers you select to create a merged bitmap must be a continuous range of two or more layers at the same level. For example, you cannot select one layer inside a group, and a layer outside that group, and merge them. Instead, you must select the entire group and the separate layer.

Setting publish options

The Publish settings in the PSD Importer let you specify the degree of compression and document quality to apply to the image when publishing the Flash document as a SWF file. These settings take effect only when you publish the document as a SWF file, and have no effect on the image when you import it to the Flash Stage or library.

Compression Lets you choose either lossy or lossless compression formats:

- **Lossy** Lossy (JPEG) compresses the image in JPEG format. To use the default compression quality specified for the imported image, select Use Publish Setting. To specify a new quality compression setting, select the Custom option, and enter a value between 1 and 100 in the Quality text field. (A higher setting preserves greater image integrity but yields a larger file size.).

- **Lossless** Lossless (PNG/GIF) compresses the image with lossless compression, in which no data is discarded from the image.

***Note:** Use Lossy compression for images with complex color or tonal variations, such as photographs or images with gradient fills. Use Lossless compression for images with simple shapes and relatively few colors.*

Calculate Bitmap Size Determines the number of bitmaps that are created for a given layer based on your import selections, and the compressed size in kilobytes of the resulting bitmaps on the layer. For example, if you select a layer with a drop shadow and a blur, and maintain layer styles, the Calculate Bitmap Size information states that three bitmaps will result from the import—one for each filter effect, and another for the image itself. To calculate the size of all the bitmaps to be imported, select all of the layers and click Calculate Bitmap Size.

Imported bitmaps

Working with imported bitmaps

When you import a bitmap into Flash, you can modify that bitmap and use it in your Flash document in a variety of ways.

If a Flash document displays an imported bitmap at a size larger than the original, the image may be distorted. To be sure that images are displayed properly, preview imported bitmaps.

Use the Property inspector to work with bitmaps

When you select a bitmap on the Stage, the Property inspector displays the bitmap's symbol name and its pixel dimensions and position on the Stage. Using the Property inspector, you can *swap* an instance of a bitmap—that is, replace the instance with an instance of another bitmap in the current document.

Display the Property inspector with bitmap properties

1. Select an instance of a bitmap on the Stage.
2. Select Window > Properties > Properties.

Replace an instance of a bitmap with an instance of another bitmap

1. Select a bitmap instance on the Stage.
2. Select Window > Properties > Properties, and click Swap.
3. Select a bitmap to replace the one currently assigned to the instance.

Set bitmap properties

You can apply anti-aliasing to an imported bitmap to smooth the edges in the image. You can also select a compression option to reduce the bitmap file size and format the file for display on the web.

1. Select a bitmap in the Library panel and click the Properties button at the bottom of the Library panel.
2. Select Allow Smoothing.

3 For Compression, select one of the following options:

Photo (JPEG) Compresses the image in JPEG format. To use the default compression quality specified for the imported image, select Use Document Default Quality. To specify a new quality compression setting, deselect Use Document Default Quality and enter a value between 1 and 100 in the Quality text field. (A higher setting preserves greater image integrity but yields a larger file size.)

Lossless (PNG/GIF) Compresses the image with lossless compression, in which no data is discarded from the image.

Note: *Use Photo compression for images with complex color or tonal variations, such as photographs or images with gradient fills. Use Lossless compression for images with simple shapes and relatively few colors.*

4 To determine the results of the file compression, click Test. To determine if the selected compression setting is acceptable, compare the original file size to the compressed file size.

5 Click OK.

Note: *JPEG Quality settings that you select in the Publish Settings dialog box do not specify a quality setting for imported JPEG files. Specify a quality setting for each imported JPEG file in the Bitmap Properties dialog box.*

Import bitmaps at runtime

To add bitmaps to a document at runtime, use the ActionScript™ 2.0 or the ActionScript 3.0 `BitmapData` command. To do so, specify a linkage identifier for the bitmap. For more information, see Assigning linkage to assets in the library in *Learning ActionScript 2.0 in Adobe Flash* or Exporting library symbols for ActionScript in *Programming ActionScript 3.0*.

1 Select the bitmap in the Library panel.

2 Do one of the following:

- Select Linkage from the Panel menu in the upper-right corner of the panel.
- Right-click (Windows) or Control-click (Macintosh) the bitmap name in the Library panel, and select Properties from the context menu.

If the Linkage properties aren't visible in the Properties dialog box, click Advanced.

3 For Linkage, select Export For ActionScript.

4 Enter an identifier string in the text field, and click OK.

Apply a bitmap as a fill

To apply a bitmap as a fill to a graphic object, use the Color panel. Applying a bitmap as a fill tiles the bitmap to fill the object. The Gradient Transform tool allows you to scale, rotate, or skew an image and its bitmap fill.

1 To apply the fill to existing artwork, select a graphic object or objects on the Stage.

2 Select Window > Color.

3 Select Bitmap from the pop-up menu in the center of the panel.

4 To use a larger preview window to display more bitmaps in the current document, click the arrow in the lower-right corner to expand the Color panel.

5 Click a bitmap to select it.

The bitmap becomes the current fill color. If you selected artwork in step 1, the bitmap is applied as a fill to the artwork.

See also

"Transform gradient and bitmap fills" on page 190

Edit bitmaps in an external editor

If you are editing a Fireworks PNG file imported as a flattened image, edit the PNG source file for the bitmap, when available.

***Note:** You cannot edit bitmaps from Fireworks PNG files imported as editable objects in an external image editor.*

If you have Fireworks 3 or later or another image-editing application installed on your system, you can start the application from Flash to edit an imported bitmap.

Edit a bitmap with Fireworks 3 or later

1. In the Library panel, right-click (Windows) or Control-click (Macintosh) the bitmap's icon and select Edit With Fireworks 3.
2. Specify whether to open the PNG source file or the bitmap file.
3. Perform the desired modifications to the file in Fireworks.
4. In Fireworks, select File > Update.
5. Return to Flash.

The file is automatically updated in Flash.

Edit a bitmap with another image-editing application

1. In the Library panel, right-click (Windows) or Control-click (Macintosh) the bitmap's icon, and select Edit With.
2. Select an image-editing application to open the bitmap file, and click OK.
3. Perform the desired modifications to the file in the image-editing application.
4. Save the file in the image-editing application.

The file is automatically updated in Flash.

5. Return to Flash to continue editing the document.

Break apart a bitmap

Breaking apart a bitmap separates the pixels in the image into discrete areas that can be selected and modified separately. When you break apart a bitmap, you can modify the bitmap with the Flash drawing and painting tools. Using the Lasso tool with the Magic Wand modifier, you can select areas of a bitmap that are broken apart.

To paint with a broken-apart bitmap, select the bitmap with the Eyedropper tool and apply the bitmap as a fill with the Paint Bucket tool or another drawing tool.

See also

"Use the Stroke Color and Fill Color controls in the Tools panel" on page 186

Break a bitmap apart

1. Select a bitmap in the current scene.
2. Select Modify > Break Apart.

Change the fill of areas of a broken-apart bitmap

1 Select the Lasso tool, click the Magic Wand modifier, and set the following options:.

- For Threshold, enter a value between 1 and 200 to define how closely the color of adjacent pixels must match to be included in the selection. A higher number includes a broader range of colors. If you enter 0, only pixels of the exact same color as the first pixel you click are selected.
- For Smoothing, select an option to define how much the edges of the selection are smoothed.

2 To select an area, click the bitmap. To add to the selection, continue clicking.

3 To fill the selected areas in the bitmap, select the fill to use.

4 To apply the new fill, select the Paint Bucket tool and click anywhere in the selected area.

Use the Eyedropper tool to apply a fill

1 Select the Eyedropper tool and click the broken-apart bitmap on the Stage. The Eyedropper tool sets the bitmap to be the current fill and changes the active tool to the Paint Bucket.

2 Do one of the following:

- To apply the bitmap as a fill, click an existing graphic object with the Paint Bucket tool.
- Select the Oval, Rectangle, or Pen tool, and draw a new object. The object is filled with the broken-apart bitmap.

To scale, rotate, or skew the bitmap fill, use the Paint Bucket tool.

Convert bitmaps to vector graphics

The Trace Bitmap command converts a bitmap into a vector graphic with editable, discrete areas of color. You manipulate the image as a vector graphic, and you can reduce file size.

When you convert a bitmap to a vector graphic, the vector graphic is no longer linked to the bitmap symbol in the Library panel.

***Note:** If the imported bitmap contains complex shapes and many colors, the converted vector graphic might have a larger file size than the original bitmap. To find a balance between file size and image quality, try a variety of settings in the Trace Bitmap dialog box.*

You can also break apart a bitmap to modify the image with Flash drawing and painting tools.

1 Select a bitmap in the current scene.

2 Select Modify > Bitmap > Trace Bitmap.

3 Enter a Color Threshold value.

When two pixels are compared, if the difference in the RGB color values is less than the color threshold, the two pixels are considered the same color. As you increase the threshold value, you decrease the number of colors.

4 For Minimum Area, enter a value to set the number of surrounding pixels to consider when assigning a color to a pixel.

5 For Curve Fit, select an option to determine how smoothly outlines are drawn.

6 For Corner Threshold, select an option to determine whether sharp edges are retained or smoothed out.

To create a vector graphic that looks most like the original bitmap, enter the following values:

- Color Threshold: 10
- Minimum Area: 1 pixel
- Curve Fit: Pixels
- Corner Threshold: Many Corners

Chapter 6: Drawing

The drawing tools in Adobe® Flash® CS3 Professional let you create and modify shapes for the artwork in your documents.

Before you draw and paint in Adobe® Flash® CS3 Professional, it is important to understand how Adobe® Flash® CS3 Professional creates artwork, how drawing tools work, and how drawing, painting, and modifying shapes can affect other shapes on the same layer.

Drawing Basics

About drawing

The drawing tools in Adobe® Flash® CS3 Professional let you create and modify shapes for the artwork in your documents.

For a text tutorial about drawing, see "Draw in Flash" on the Flash Tutorials page at www.adobe.com/go/learn_fl_tutorials.

For a video tutorial about drawing, see www.adobe.com/go/vid0119.

About vector and bitmap graphics

Computers display graphics in either vector or bitmap format. Understanding the difference between the two formats helps you work more efficiently. Using Flash, you can create and animate compact vector graphics. Flash also imports and manipulates vector and bitmap graphics that were created in other applications.

Vector graphics

Vector graphics describe images by using lines and curves, called *vectors,* that also include color and position properties. For example, the image of a leaf is described by points through which lines pass, creating the leaf's outline. The color of the leaf is determined by the color of the outline and the color of the area enclosed by the outline.

Lines in vector art

When you edit a vector graphic, you modify the properties of the lines and curves that describe its shape. Move, resize, reshape, and change the color of a vector graphic without changing the quality of its appearance. Vector graphics are resolution independent; that is, they can be displayed on output devices of varying resolutions without losing any quality.

Bitmap graphics

Bitmap graphics describe images using colored dots, called *pixels,* arranged in a grid. For example, the image of a leaf 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.

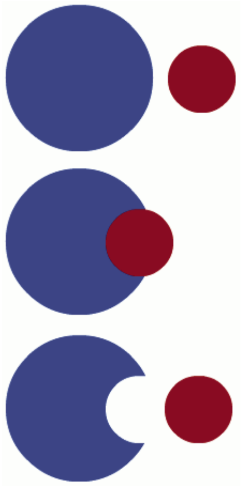
Pixels in bitmap art

When you edit a bitmap graphic, you modify pixels rather than lines and curves. Bitmap graphics are resolution dependent, because the data describing the image is fixed to a grid of a particular size. Editing a bitmap graphic can change the quality of its appearance. In particular, resizing a bitmap graphic can make the edges of the image ragged as pixels are redistributed within the grid. Displaying a bitmap graphic on an output device that has a lower resolution than the image itself also degrades its quality.

About the Flash drawing models

Flash provides two drawing models that give you a great deal of flexibility when drawing shapes:

Merge Drawing model The default drawing model automatically merges shapes that you draw when you overlap them. If you select a shape that is merged with another, and move it, the shape below it is permanently altered. For example, if you draw a circle and overlay a smaller circle on top of it, and then select the circle and move it, the portion of the second circle that overlaid the first circle is removed.

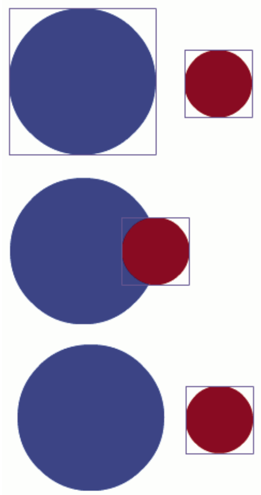


Shapes created with the merge drawing mode merge together when they overlap. Selecting a shape and moving it results in the overlaid shape being altered.

Object Drawing model Draws shapes as separate objects that do not automatically merge together when overlaid. This overlaps shapes without altering their appearance if you move them apart, or rearrange their appearance. Flash creates each shape as a separate object that you can individually manipulate.

When you select a shape created using the Object Drawing model, Flash surrounds the shape with a rectangular bounding box. Use the Pointer tool to move the object by clicking the bounding box and dragging the shape to position it on the Stage.

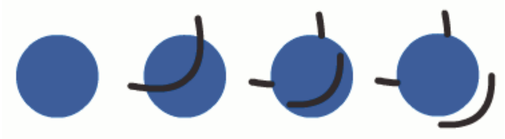
Note: *Set preferences for contact sensitivity when selecting shapes created using the Object Drawing model.*



Shapes created with the Object Drawing model remain as separate objects the you can individually manipulate.

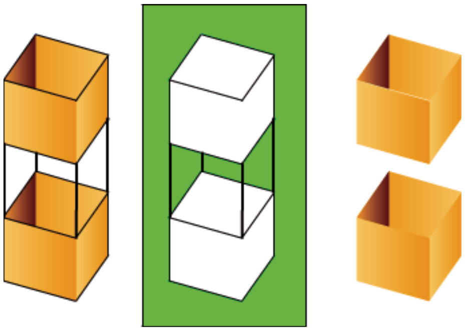
About overlapping shapes

When you use the Pencil, Pen, Line, Oval, Rectangle, or Brush tool to draw a line across another line or painted shape, the overlapping lines are divided into segments at the intersection points. To select, move, and reshape each segment individually, use the Selection tool.



A fill; the fill with a line drawn through it; and the three line segments created by segmentation

When you paint on top of shapes and lines, the portion underneath is replaced by whatever is on top. Paint of the same color merges together. Paint of different colors remains distinct. To create masks, cutouts, and other negative images, use these features. For example, the following cutout is made by moving the ungrouped kite image onto the green shape, deselecting the kite, and then moving the filled portions of the kite away from the green shape.



Making a cutout with the kite image

To avoid inadvertently altering shapes and lines by overlapping them, group the shapes or use layers to separate them.

See also

“Group objects” on page 201

“About layers” on page 36

Use the Object Drawing model

By default, Flash uses the Merge Drawing model. To draw shapes using the Object Drawing model, click the Object Drawing button in the Tools panel.

Enable the object drawing model

1 Select a drawing tool that supports the Object Drawing model (the Pencil, Line, Pen, Brush, Oval, Rectangle, and Polygon tools).

2 Select the Object Drawing button ○ from the Options category of the Tools panel, or press the J key to toggle between the Merge and Object Drawing models. The Object Drawing button toggles between the Merge and Object Drawing models. Set preferences for contact sensitivity when you select shapes created by using the Object Drawing model.

Convert a Merge Drawing model created shape to an Object Drawing model shape

1 Select the shape on the Stage.

2 To convert the shape into a Object Drawing model shape, select Modify > Combine Objects > Union. After conversion, the shape is treated like a vector-based drawing object that does not alter its appearance by interacting with other shapes.

To join two or more shapes into a single, object-based shape, use the Union command.

Select objects

Select objects with the Pointer, Subselection, and Lasso tools.

The Pointer, Subselection, and Lasso tools select objects by clicking on them. The Pointer and Subselection tools select objects by dragging a rectangular selection marquee around the object. The Lasso tool selects objects by dragging a free-form selection marquee around the object. When an object is selected, a rectangular box appears around the object.

1 Select Edit > Preferences (Windows) or Flash > Preferences (Macintosh).

2 In the General category of the Preferences dialog box, do one of the following:

- To select only objects and points that are completely enclosed by the selection marquee, deselect Contact-Sensitive Selection and Lasso tools. Points that lie within the selection area will still be selected.
- To select objects or groups that are only partially enclosed by the selection marquee, select Contact-Sensitive Selection and Lasso tools.

Combining objects

To create new shapes by combining or altering existing objects, use the Combine Objects commands in the Modify menu (Modify > Combine Objects). In some cases, the stacking order of selected objects determines how the operation works. The Combine Objects commands are:

Union Joins two or more shapes into a single shape. The result is a single Object Drawing model shape consisting of all the portions visible on the shapes before they were unified. The unseen, overlapping portions of the shapes are deleted.

Note: *Unlike when you use the Group command (Modify > Group), you cannot break apart shapes joined by using the Union command.*

Intersect Creates an object from the intersection of two or more objects. The resulting Object Drawing shape consists of the overlapping portions of the combined shapes. Any part of the shape that doesn't overlap is deleted. The resulting shape uses the fill and stroke of the top-most shape in the stack.

Punch Removes portions of a selected object as defined by the overlapping portions of another selected object positioned in front of it. Any part of a shape that is overlapped by the top-most shape is deleted, and the top-most shape is deleted entirely. The resulting shapes remain as separate objects, and are not combined into a single object (unlike the Union or Intersect commands, which join the objects together).

Crop Uses the shape of one object to crop another object. The front or topmost object defines the shape of the cropped area. Any part of an underlying shape that overlaps with the top-most shape remains, while all other portions of the underlying shapes are deleted, and the top-most shape is deleted entirely. The resulting shapes remain as separate objects, and are not combined into a single object (unlike the Union or Intersect commands, which join the objects).

Specify drawing preferences

Set drawing settings to specify snapping, smoothing, and straightening behaviors. Change the tolerance setting for each option, and turn each option off or on. Tolerance settings are relative, depending on the resolution of your computer screen and the current magnification of the scene. By default, each option is on and set to Normal tolerance.

Set drawing settings

1 Select Edit > Preferences (Windows) or Flash > Preferences (Macintosh) and select Editing.

2 Under Drawing Settings, select from the following options:

Connect Lines Determines how close the end of a line being drawn must be to an existing line segment before the end point snaps to the nearest point on the other line. This setting also controls horizontal and vertical line recognition—that is, how nearly horizontal or vertical a line must be drawn before Flash makes it exactly horizontal or vertical. When Snap To Objects is turned on, this setting controls how close objects must be to snap to one another.

Smooth Curves Specifies the amount of smoothing applied to curved lines drawn with the Pencil tool when the drawing mode is set to Straighten or Smooth. (Smoother curves are easier to reshape, whereas rougher curves match the original line strokes more closely.)

***Note:** To further smooth existing curved segments, use Modify > Shape > Smooth and Modify > Shape > Optimize.*

Recognize Lines Defines how nearly straight a line segment drawn with the Pencil tool must be before Flash recognizes it as a straight line and makes it perfectly straight. If Recognize Lines is off while you draw, straighten lines later by selecting one or more line segments and selecting Modify > Shape > Straighten.

Recognize Shapes Controls how precisely to draw circles, ovals, squares, rectangles, and 90- and 180-degree arcs for them to be recognized as geometric shapes and redrawn accurately. The options are Off, Strict, Normal, and Tolerant. Strict requires that the shape be drawn very close to straight; Tolerant specifies that the shape can be somewhat rough, and Flash will redraw the shape. If Recognize Shapes is off while you draw, straighten lines later by selecting one or more shapes (for example, connected line segments) and selecting Modify > Shape > Straighten.

Click Accuracy Specifies how close to an item the pointer must be before Flash recognizes the item.

Specify the Selection, Subselection, and Lasso tool contact-sensitivity options when you create shapes using the Object Drawing model. By default, objects are only selected when the tool's marquee rectangle completely surrounds the object. Deselecting this option selects entire objects when they are only partially enclosed by the selection marquee of the Selection, Subselection, or Lasso tools.

Set Selection, Subselection, and Lasso tool contact options

1 Select Edit > Preferences (Windows) or Flash > Preferences (Macintosh).

2 In the General category, do one of the following:

- To select only objects and points that are completely enclosed by the selection marquee, deselect Contact-Sensitive Selection and Lasso tools. Points that lie within the selection area are still selected.
- To select objects or groups that are only partially enclosed by the selection marquee, select Contact-Sensitive Selection and Lasso tools.

***Note:** The Subselection tools use the same contact-sensitive setting.*

Using Flash drawing and painting tools

About Flash drawing and painting tools

When you use most Flash tools, the Property inspector changes to present the settings associated with that tool. For example, if you select the Text tool, the Property inspector displays text properties, making it easy to select the desired text attributes.

When you use a drawing or painting tool to create an object, the tool applies the current stroke and fill attributes to the object. To change the stroke and fill attributes of existing objects, use the Paint Bucket and Ink Bottle tools in the Tools panel or the Property inspector.

Reshape lines and shape outlines in a variety of ways after you create them. Fills and strokes are treated as separate objects. Select fills and strokes separately to move or modify them.

To automatically align elements with each other and with the drawing grid or guides, use snapping.

To change the display of tools, customize the Tools panel.

See also

“Using Flash authoring panels” on page 40

“Use the Stroke Color and Fill Color controls in the Tools panel” on page 186

“Use the Stroke Color and Fill Color controls in the Property inspector” on page 186

“Reshaping lines and shape outlines” on page 177

“About the main toolbar and edit bar” on page 26

“Use the Tools panel” on page 26

Draw with the Pencil tool

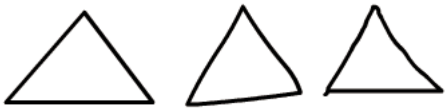
To draw lines and shapes, use the Pencil tool, in much the same way that you use a real pencil to draw. To apply smoothing or straightening to the lines and shapes as you draw, select a drawing mode for the Pencil tool.

1 Select the Pencil tool .

2 Select Window > Properties > Properties and select a stroke color, line weight, and style.

3 Select a drawing mode under Options in the Tools panel:

- To draw straight lines and convert approximations of triangles, ovals, circles, rectangles, and squares into these common geometric shapes, select Straighten .
- To draw smooth curved lines, select Smooth .
- To draw freehand lines with no modification applied, select Ink .



Lines drawn with Straighten, Smooth, and Ink mode, respectively.

4 To draw with the Pencil tool, Shift-drag to constrain lines to vertical or horizontal directions, click the Stage, and drag.

See also

“Use the Stroke Color and Fill Color controls in the Property inspector” on page 186

Draw straight lines

To draw one straight line segment at a time, use the Line tool.

1 Select the Line tool .

2 Select Window > Properties > Properties and select stroke attributes.

***Note:** You cannot set fill attributes for the Line tool.*

3 Click the Object Drawing button in the Options section of the Tools panel, to select either the Merge or Object drawing model. When the Object Drawing button is depressed, the Line tool is in Object drawing mode.

4 Position the pointer where the line is to begin, and drag to where the line is to end. To constrain the angle of the line to multiples of 45 degrees, Shift-drag.

See also

"Use the Stroke Color and Fill Color controls in the Property inspector" on page 186

"About the Flash drawing models" on page 160

Draw rectangles and ovals

The Oval and Rectangle tools let you create these basic geometric shapes, and apply strokes, fills, and specify rounded corners. In addition to the Merge and Object drawing model, the Oval and Rectangle tools also provide the Primitive Object drawing mode.

When you create rectangles or ovals using the Primitive Rectangle or Primitive Oval tools, Flash draws the shapes as separate objects not unlike the shapes created when you use object drawing mode. The Primitive shape tools let you specify the corner radius of rectangles, and the start and end angle and the inner radius of ovals using controls in the Property inspector. After you create a primitive shape, alter the radiuses and dimensions by selecting the shape on the Stage, and adjusting the controls in the Property inspector.

***Note:** When either of the Primitive Object drawing tools is selected, the Property inspector retains the values of the last primitive object that you edited. For example, if you modify a rectangle, and then draw a second rectangle.*

See also

"Use the Stroke Color and Fill Color controls in the Property inspector" on page 186

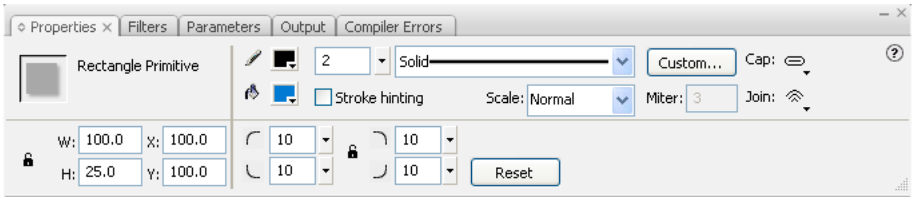
Draw primitive rectangles

1 To select the Rectangle Primitive tool, click and hold the mouse button on the Rectangle tool , and select the Rectangle Primitive tool from the pop-up menu.

2 To create a primitive rectangle, drag the Rectangle Primitive tool on the Stage.

***Note:** To change the corner radius while dragging with the Primitive Rectangle tool, press the Up Arrow key or Down Arrow key. When the corners achieve the desired roundness, release the key.*

3 With the primitive rectangle selected on the Stage, you can use the controls found in the Property inspector to further modify the shape, or specify fill and stroke colors.



2
Solid
☐ Stroke hinting
Scale: Normal
Miter: 3
W: 100.0 X: 100.0
H: 25.0 Y: 100.0
10 10
10 10

Properties for a rectangle primitive

The Property inspector controls specific to the Rectangle Primitive tool are:

Rectangle Corner Radius Controls Lets you specify the corner radiuses for the rectangle. You can enter a numerical value for the inner radius in the box, or click the slider and interactively adjust the size of the radius. Entering a negative value creates an inverse radius. You can also deselect the constrain corner radius icon, and adjust each corner radius individually.

Reset Resets all of the Rectangle Primitive tool's controls, and returns the primitive rectangle shape drawn on the stage to its initial size and shape.

4 To specify a different corner radius for each corner of the rectangle, deselect the Lock icon located in the Primitive Rectangle radius controls sections of the Property inspector. When locked, the radius controls are restrained so that each corner uses the same radius.

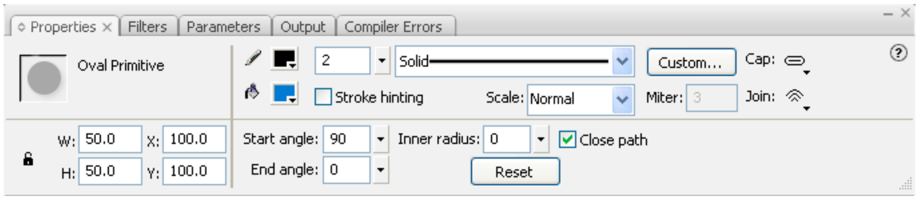
5 To reset the corner radiuses, click the Reset button in the Property inspector.

Draw primitive ovals

1 Click and hold the mouse button on the Rectangle tool , and select the Oval Primitive tool .

2 To create a primitive oval, drag the Primitive Oval tool on the Stage. To constrain the shape to a circle, Shift-drag.

3 With the primitive oval selected on the Stage, you can use the controls found in the Property inspector to further modify the shape, or specify fill and stroke colors.



Oval Primitive

2 Solid

□ Stroke hinting Scale: Normal Miter: 3

W: 50.0 X: 100.0

H: 50.0 Y: 100.0

Start angle: 90 Inner radius: 0 ☑ Close path

End angle: 0

Properties for an oval primitive

The Property inspector controls specific to the Primitive Oval tool are:

Start angle and **End angle** Let you specify the angle of the start and end points of the oval. Using these controls, you can easily modify the shape of ovals and circles into pie slices, half circles, and other creative shapes.

Inner radius Lets you specify an inner radius (or oval) within the oval. You can either enter a numerical value for the inner radius in the box, or click the slider and interactively adjust the size of the inner radius. The numerical values you can enter for the inner radius range from 0 to 99, and represent the percentage of the oval's fill that is removed.

Close path Lets you specify whether the path (or paths, if you are specifying an inner radius) of the oval is closed. If you specify an open path, no fill is applied to the resulting shape, only the stroke is drawn. Close path is selected by default.

Reset Resets all of the Oval Primitive tool's controls, and returns the primitive oval shape drawn on the stage to its initial size and shape.

Draw ovals and rectangles

The Oval and Rectangle tools create these basic geometric shapes.

1 To select the Rectangle tool or Oval tool , click and hold the mouse button on the Rectangle tool and drag.

2 To create a rectangle or oval, drag the Rectangle or Oval tool on the Stage.

3 For the Rectangle tool, specify rounded corners by clicking the Round Rectangle modifier and entering a corner radius value. A value of zero (0) creates square corners.

4 Drag on the Stage. If you are using the Rectangle tool, press the Up Arrow and Down Arrow keys while dragging to adjust the radius of rounded corners.

For the Oval and Rectangle tools, Shift-drag to constrain the shapes to circles and squares.

To specify a specific size of Oval or Rectangle in pixels, press the Alt key (Windows) or Option key (Macintosh) with the Oval or Rectangle tool selected, and click the Stage to display the Oval And Rectangle Settings dialog box.

- For ovals, specify the width and height in pixels, and whether to draw the oval from the center.
- For rectangles, specify the width and height in pixels, the radius of the rectangle corners for rounded corners, and whether to draw the rectangle from the center.

Draw polygons and stars

The PolyStar tool draws polygons or stars.

1 To select the PolyStar tool from the pop-up menu, click and hold the mouse button on the Rectangle tool and drag.

2 Select Window > Properties > Properties and select stroke and fill attributes.

3 Click Options and do the following:

- For Style, select Polygon or Star.
- For Number Of Sides, enter a number between 3 and 32.
- For Star Point Size, enter a number between 0 and 1 to specify the depth of the star points. A number closer to 0 creates deeper points (like needles). If you are drawing a polygon, leave this setting unchanged. (It does not affect the polygon shape.)

4 Click OK.

5 Drag on the Stage.

See also

"Use the Stroke Color and Fill Color controls in the Property inspector" on page 186

Paint with the Brush tool

The Brush tool draws brush-like strokes, as if you were painting. It creates special effects, including calligraphic effects. Select a brush size and shape using the Brush tool modifiers.

Brush size for new strokes remains constant even when you change the magnification level for the Stage, so the same brush size appears larger when the Stage magnification is lower. For example, suppose you set the Stage magnification to 100% and paint with the Brush tool using the smallest brush size. Then, you change the magnification to 50% and paint again using the smallest brush size. The new stroke that you paint appears 50% thicker than the earlier stroke. (Changing the magnification of the Stage does not change the size of existing brush strokes.)

Use an imported bitmap as a fill when painting with the Brush tool. See "Break apart groups and objects" on page 202.

If you have a Wacom pressure-sensitive tablet connected to your computer, vary the width and angle of the brush stroke by using the Brush tool Pressure and Tilt modifiers, and varying pressure on the stylus.

The Pressure modifier varies the width of brush strokes when you vary the pressure on the stylus. The Tilt modifier varies the angle of brush strokes when you vary the angle of the stylus on the tablet. The Tilt modifier measures the angle between the top (eraser) end of the stylus and the top (north) edge of the tablet. For example, if you hold the pen vertically against the tablet, the Tilt is 90. The Pressure and Tilt modifiers are both fully supported for the eraser function of the stylus.

A variable-width brush stroke drawn with a stylus

1 Select the Brush tool 🖌 .

2 Select Window > Properties > Properties and select a fill color.

3 Click the Brush Mode modifier and select a painting mode:

Paint Normal Paints over lines and fills on the same layer.

Paint Fills Paints fills and empty areas, leaving lines unaffected.

Paint Behind Paints in blank areas of the Stage on the same layer, leaving lines and fills unaffected.

Paint Selection Applies a new fill to the selection when you select a fill in the Fill Color control or the Fill box of the Property inspector, the same as selecting a filled area and applying a new fill.

Paint Inside Paints the fill in which you start a brush stroke and never paints lines. If you start painting in an empty area, the fill doesn't affect any existing filled areas.

4 Select a brush size and brush shape from the Brush tool modifiers.

5 If a Wacom pressure-sensitive tablet is attached to your computer, select the Pressure modifier, the Tilt modifier, or both, to modify brush strokes.

- Select the Pressure modifier to vary the width of your brush strokes by varying the pressure on your stylus.
- To vary the angle of your brush strokes by varying the angle of the stylus on the Wacom pressure-sensitive tablet, select the Tilt modifier.

6 Drag on the Stage. To constrain brush strokes to horizontal and vertical directions, Shift-drag.

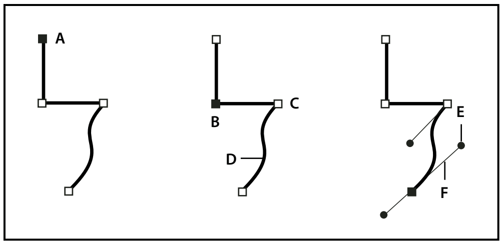
See also

"Use the Stroke Color and Fill Color controls in the Property inspector" on page 186

About paths

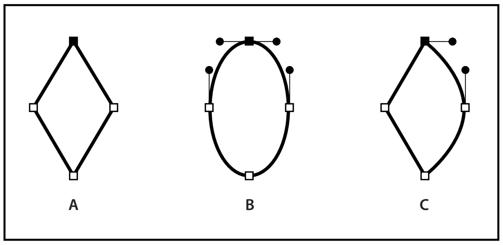
Whenever you draw a line or shape in Flash, you create a line called a *path*. A path is made up of one or more straight or curved *segments*. The beginning and end of each segment are marked by *anchor points*, which work like pins holding a wire in place. A path can be *closed* (for example, a circle), or *open*, with distinct *endpoints* (for example, a wavy line).

You change the shape of a path by dragging its anchor points, the *direction points* at the end of *direction lines* that appear at anchor points, or the path segment itself.



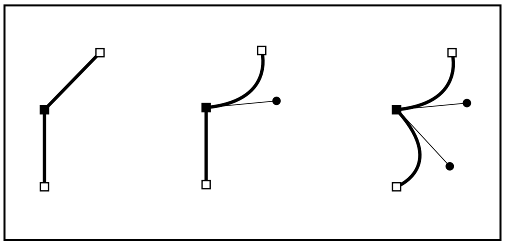
Components of a path
***A.** Selected (solid) endpoint **B.** Selected anchor point **C.** Unselected anchor point **D.** Curved path segment **E.** Direction line **F.** Direction point*

Paths can have two kinds of anchor points: corner points and smooth points. At a *corner point*, a path abruptly changes direction. At a *smooth point*, path segments are connected as a continuous curve. You can draw a path using any combination of corner and smooth points. If you draw the wrong kind of point, you can always change it.



Points on a path
***A.** Four corner points **B.** Four smooth points **C.** Combination of corner and smooth points*

A corner point can connect any two straight or curved segments, while a smooth point always connects two curved segments.



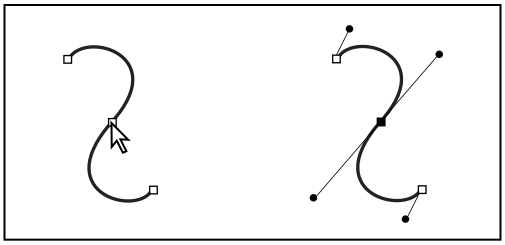
A corner point can connect both straight segments and curved segments.

***Note:** Don't confuse corner and smooth points with straight and curved segments.*

A paths outline is called a *stroke*. A color or gradient applied to an open or closed paths interior area is called a *fill*. A stroke can have weight (thickness), color, and a dash pattern. After you create a path or shape, you can change the characteristics of its stroke and fill.

About direction lines and direction points

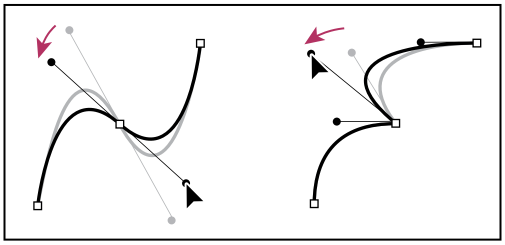
When you select an anchor point that connects curved segments (or select the segment itself), the anchor points of the connecting segments display *direction handles*, which consist of *direction lines* that end in *direction points.* The angle and length of the direction lines determine the shape and size of the curved segments. Moving the direction points reshapes the curves. Direction lines don't appear in the final output.



After selecting an anchor point (left), direction lines appear on any curved segments connected by the anchor point (right).

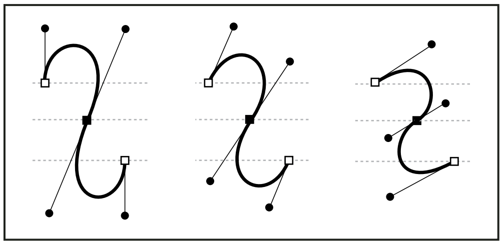
A smooth point always has two direction lines, which move together as a single, straight unit. When you move a direction line on a smooth point, the curved segments on both sides of the point are adjusted simultaneously, maintaining a continuous curve at that anchor point.

In comparison, a corner point can have two, one, or no direction lines, depending on whether it joins two, one, or no curved segments, respectively. Corner point direction lines maintain the corner by using different angles. When you move a direction line on a corner point, only the curve on the same side of the point as that direction line is adjusted.



Adjusting direction lines on a smooth point (left) and a corner point (right)

Direction lines are always tangent to (perpendicular to the radius of) the curve at the anchor points. The angle of each direction line determines the slope of the curve, and the length of each direction line determines the height, or depth, of the curve.



Moving and resizing direction lines changes the slope of curves.

Drawing with the Pen tool

Interacting with the Pen tool

To draw precise paths as straight lines or smooth, flowing curves, use the Pen tool. When you draw with the Pen tool, click to create points on straight line segments, and drag to create points on curved line segments. Adjust straight and curved line segments by adjusting points on the line. Convert curves to straight lines and the reverse, and display points on lines that you create with other Flash drawing tools, such as the Pencil, Brush, Line, Oval, or Rectangle tools, to adjust those lines.

The Pen tool provides feedback about its current drawing state by displaying different pointers. The various drawing states are indicated by the following pointers:

Initial Anchor Point pointer The first pointer you see when you select the Pen tool. Indicates that the next mouse click on the stage will create an initial anchor point, which is the beginning of a new path (all new paths begin with an initial anchor point). Any existing drawing paths are terminated.

Sequential Anchor Point pointer Indicates that the next mouse click will create an anchor point with a line connecting it to the previous anchor point. This pointer is displayed during the creation of all user-defined anchor points except the initial anchor point of a path.

Add Anchor Point pointer Indicates that the next mouse click will add an anchor point to an existing path. To add an anchor point, the path must be selected, and the Pen tool must not be over an existing anchor point. The existing path is redrawn based on the additional anchor point. Only one anchor point can be added at a time.

Delete Anchor Point pointer Indicates that the next mouse click on an existing path will remove an anchor point. To remove an anchor point, the path must be selected with the Selection tool, and the pointer must be over an existing anchor point. The existing path is redrawn based on the removal of the anchor point. Only one anchor point can be removed at a time.

Continue Path pointer Extends a new path from an existing anchor point. For this pointer to be activated, the mouse must be over an existing anchor point on a path. This pointer is only available when you are not currently drawing a path. The anchor point does not have to be one of the terminal anchor points of a path; any anchor point can be the location of a continued path.

Close Path pointer Closes the path you're drawing on the starting point of the path. You can only close a path that you are currently drawing, and the existing anchor point must be the starting anchor point of the same path. The resulting path does not have any specified fill color settings applied to the enclosed shape; apply fill color separately.

Join Paths pointer Similar to the Close Path Tool except that the mouse must not be over the initial anchor point of the same path. The pointer must be over either of the terminal points of a unique path. The segment may or may not be selected.

***Note:** Joining paths may or may not result in a closed shape.*

Retract Bezier Handle pointer Appears when the mouse is over an anchor point whose Bezier handles are displayed. Clicking the mouse retracts the Bezier handles, and causes the curved path across the anchor point to revert to straight segments.

Convert Anchor Point pointer Converts a corner point without direction lines to a corner point with independent direction lines. To enable the convert anchor point pointer, use the Shift + C modifier keys to toggle the Pen tool.

For a video tutorial about the Pen tool, see www.adobe.com/go/vid0120.

See also

"Reshaping lines and shape outlines" on page 177

Set Pen tool preferences

Specify preferences for the appearance of the Pen tool pointer, for previewing line segments as you draw, and for the appearance of selected anchor points. Selected line segments and anchor points use the outline color of the layer on which the lines and points appear.

1 Select the Pen tool , then select Edit > Preferences (Windows) or Flash > Preferences (Macintosh) and click Editing.

2 In the Category list, select Drawing.

3 Set the following options for the Pen tool:

Show Pen Preview Previews line segments as you draw. A preview of the line segment appears as you move the pointer around the Stage, before you click to create the end point of the segment. If this option is not selected, a line segment does not appear until you create the end point.

Show Solid Points Displays selected anchor points as hollow and deselected anchor points as solid. If this option is not chosen, selected anchor points are solid, and deselected anchor points are hollow.

Show Precise Pointers Specifies that the Pen tool pointer appears as a cross-hair pointer, rather than the default Pen Tool icon, for more precise placement of lines. To display the default Pen Tool icon with the Pen Tool, deselect the option.

***Note:** To switch between the cross-hair pointer and the default Pen Tool icon, press the Caps Lock key.*

4 Click OK.

Draw straight lines with the Pen tool

The simplest path you can draw with the Pen Tool is a straight line, made by clicking the Pen Tool to create two anchor points. Continue to click to create a path made of straight line segments connected by corner points.

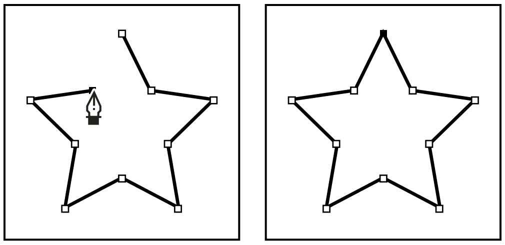
1 Select the Pen Tool .

2 Position the Pen Tool where the straight segment is to begin, and click to define the first anchor point. If direction lines appear, you accidentally dragged the Pen Tool; choose Edit > Undo and click again.

***Note:** The first segment you draw is not visible until you click a second anchor point (unless you've specified Show Pen Preview in the Drawing category of the Preferences dialog box).*

3 Click again where you want the segment to end (Shift-click to constrain the angle of the segment to a multiple of 45°).

4 Continue clicking to set anchor points for additional straight segments.



Clicking pen tool creates straight segments

5 To complete the path as an open or closed shape, do one of the following:

- To complete an open path, double-click the last point, click the Pen Tool in the Tools panel, or Control-click (Windows) or Command-click (Macintosh) anywhere away from the path.
- To close the path, position the Pen Tool over the first (hollow) anchor point. A small circle appears next to the Pen Tool pointer when it is positioned correctly. Click or drag to close the path.
- To complete the shape as is, select Edit > Deselect All, or select a different tool in the Tools panel.

See also

"Use the Stroke Color and Fill Color controls in the Property inspector" on page 186

Draw curves with the Pen tool

To create a curve, add an anchor point where a curve changes direction, and drag the direction lines that shape the curve. The length and slope of the direction lines determine the shape of the curve.

Curves are easier to edit and your system can display and print them faster if you draw them using as few anchor points as possible. Using too many points can also introduce unwanted bumps in a curve. Instead, draw widely spaced anchor points, and practice shaping curves by adjusting the length and angles of the direction lines.

1 Select the Pen tool .

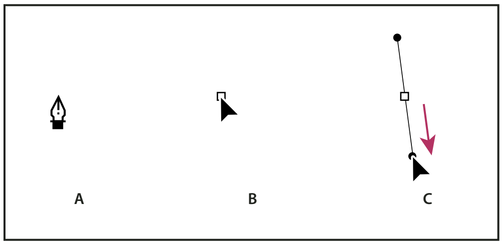
2 Position the Pen tool where the curve is to begin, and hold down the mouse button.

The first anchor point appears, and the Pen tool pointer changes to an arrowhead. (In Photoshop, the pointer changes only after you've started dragging.)

3 Drag to set the slope of the curve segment you're creating, and then release the mouse button.

In general, extend the direction line about one third of the distance to the next anchor point you plan to draw. (You can adjust one or both sides of the direction line later.)

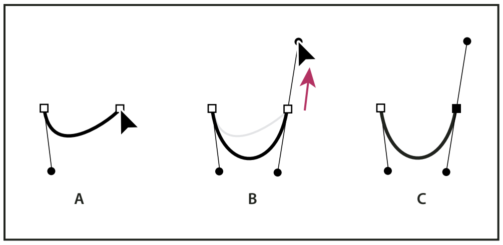
Hold down the Shift key to constrain the tool to multiples of 45˚.



Drawing the first point in a curve
***A.** Positioning Pen tool* ***B.** Starting to drag (mouse button pressed)* ***C.** Dragging to extend direction lines*

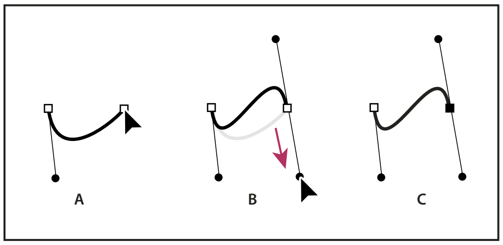
4 Position the Pen tool where the curve segment is to end, and do one of the following:

- To create a C-shaped curve, drag in a direction opposite to the previous direction line and release the mouse button.



Drawing the second point in a curve
***A.** Starting to drag second smooth point **B.** Dragging away from previous direction line, creating a C curve **C.** Result after releasing mouse button*

- To create an S-shaped curve, drag in the same direction as the previous direction line and release the mouse button.



Drawing an S curve
***A.** Starting to drag new smooth point **B.** Dragging in the same direction as previous direction line, creating an S curve **C.** Result after releasing mouse button*

5 To create a series of smooth curves, continue dragging the Pen tool from different locations. Place anchor points at the beginning and end of each curve, not at the tip of the curve.

To break out the direction lines of an anchor point, Alt-drag (Windows) or Option-drag (Macintosh) direction lines.

6 To complete the path, do one of the following:

- To close the path, position the Pen tool over the first (hollow) anchor point. A small circle appears next to the Pen tool pointer when it is positioned correctly. Click or drag to close the path.
- To leave the path open, Ctrl-click (Windows) or Command-click (Macintosh) anywhere away from all objects, select a different tool, or choose Edit > Deselect All.

Add or delete anchor points

Adding anchor points can give you more control over a path or it can extend an open path. However, it's a good idea not to add more points than necessary. A path with fewer points is easier to edit, display, and print. To reduce the complexity of a path, delete unnecessary points.

The toolbox contains three tools for adding or deleting points: the Pen tool, the Add Anchor Point tool, and the Delete Anchor Point tool.

By default, the Pen tool changes to the Add Anchor Point tool as you position it over a selected path, or to the Delete Anchor Point tool as you position it over an anchor point.

***Note:** Don't use the Delete, Backspace, and Clear keys or the Edit > Cut or Edit > Clear commands to delete anchor points; these keys and commands delete the point and the line segments that connect to that point.*

Add or delete anchor points

1 Select the path to modify.

2 Click and hold the mouse button on the Pen tool, then select the Pen tool, Add Anchor Point tool, or the Delete Anchor Point tool.

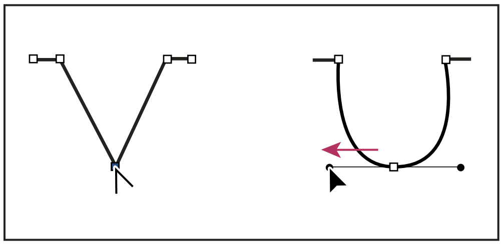
3 To add an anchor point, position the pointer over a path segment, and click. To delete an anchor point, position the pointer over an anchor point, and click.

Adjust anchor points on paths

When you draw a curve with the Pen tool, you create curve points—anchor points on a continuous, curved path. When you draw a straight line segment, or a straight line connected to a curved segment, you create corner points—anchor points on a straight path or at the juncture of a straight and a curved path.

By default, selected curve points appear as hollow circles, and selected corner points appear as hollow squares.

To convert segments in a line from straight segments to curve segments or the reverse, convert corner points to curve points or the reverse.



Dragging a direction point out of a corner point to create a smooth point

Move, add, or delete anchor points on a path. To move anchor points, use the Subselection tool to adjust the length or angle of straight segments or the slope of curved segments. Nudge selected anchor points to make small adjustments.

Deleting unneeded anchor points on a curved path optimizes the curve and reduces the file size.

- To move an anchor point, drag the point with the Subselection tool.
- To nudge an anchor point or points, select the point or points with the Subselection tool and use the arrow keys to move the point or points. Shift-click to select multiple points.
- To convert a corner point to a curve point, use the Subselection tool to select the point, then Alt-drag (Windows) or Option-drag (Macintosh) the point to place the tangent handles.
- To convert a curve point to a corner point, click the point with the Pen tool. The carat ^ marker next to the pointer indicates when it is over the curve point.
- To add an anchor point, click a line segment with the Pen tool. A plus (+) sign appears next to the Pen tool if an anchor point can be added to the selected line segment. If the line segment is not yet selected, click it with the Pen tool to select it, and then add an anchor point.
- To delete a corner point, click the point once with the Pen tool. A minus (-) sign appears next to the Pen tool if an anchor point can be deleted from the selected line segment. If the line segment is not yet selected, click it with the Pen tool to select it, and then delete the anchor point.

- To delete a curve point, click the point once with the Pen tool. A minus (-) sign appears next to the Pen tool if an anchor point can be deleted from the selected line segment. If the line segment is not yet selected, click it with the Pen tool to select it, and then delete the corner point. (Click once to convert the point to a corner point, and once more to delete the point.)
- Select the point with the Subselection tool and press Delete.

Adjust segments

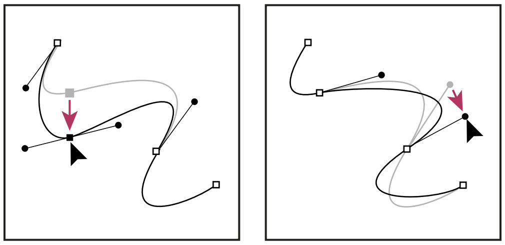
To change the angle or length of the segment, or adjust curved segments to change the slope or direction of the curve, adjust straight segments.

When you move a tangent handle on a curve point, the curves on both sides of the point adjust. When you move a tangent handle on a corner point, only the curve on the same side of the point as the tangent handle adjusts.

- To adjust a straight segment, select the Subselection tool, and select a straight segment. Use the Subselection tool to drag an anchor point on the segment to a new position.
- To adjust a curve segment, select the Subselection tool and drag the segment.

***Note:** When you click the path, Flash shows the anchor points. Adjusting a segment with the Subselection tool can add points to the path.*

- To adjust points or tangent handles on a curve, select the Subselection tool, and select an anchor point on a curved segment.
- To adjust the shape of the curve on either side of the anchor point, drag the anchor point, or drag the tangent handle. To constrain the curve to multiples of 45°, shift-drag. To drag tangent handles individually, Alt-drag (Windows) or Option-drag (Macintosh).



Drag the anchor point, or drag the direction point.

Reshaping lines and shape outlines

Display and adjust points with the Subselection tool

1 Select the Subselection tool.

2 Click the line or shape outline.

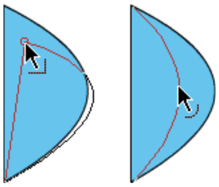
See also

"Adjust anchor points on paths" on page 176

Reshape using the Selection tool

To reshape a line or shape outline, drag any point on a line using the Selection tool. The pointer changes to indicate what type of reshaping it can perform on the line or fill.

Flash adjusts the curve of the line segment to accommodate the new position of the moved point. If the repositioned point is an end point, lengthen or shorten the line. If the repositioned point is a corner, the line segments forming the corner remain straight as they become longer or shorter.



When a corner appears next to the pointer, you can change an end point. When a curve appears next to the pointer, you can adjust a curve.

Some brush stroke areas are easier to reshape if you view them as outlines.

If you are having trouble reshaping a complex line, smooth it to remove some of its details, making reshaping easier. Increasing the magnification can also make reshaping easier and more accurate.

1 Select the Selection tool .

2 Do one of the following:

- To reshape the segment, drag from any point.
- To drag a line to create a new corner point, control-click (Windows) or Option-click (Macintosh).

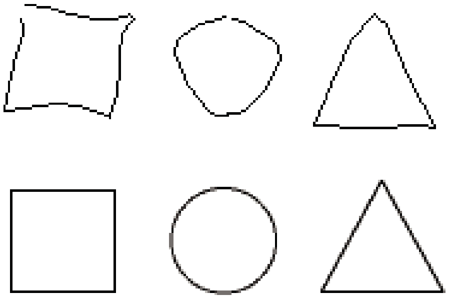
Straighten and smooth lines

To reshape lines and shape outlines, straighten or smooth them.

Note: *To adjust the degree of automatic smoothing and straightening, specify preferences for drawing settings.*

Straightening makes small straightening adjustments to lines and curves you already drew. It has no effect on segments that are already straight.

To make Flash recognize shapes, use the straightening technique. If you draw any oval, rectangular, or triangular shapes with the Recognize Shapes option turned off, use the Straightening option to make the shapes geometrically perfect. Shapes that are touching, and thus connected to other elements, are not recognized.



Shape recognition turns the top shapes into the bottom shapes.

Smoothing softens curves and reduces bumps or other variations in a curve's overall direction. It also reduces the number of segments in a curve. Smoothing is relative, however, and has no effect on straight segments. It is particularly useful when you are having trouble reshaping a number of very short curved line segments. Selecting all the segments and smoothing them reduces the number of segments, producing a gentler curve that is easier to reshape.

Repeated application of smoothing or straightening makes each segment smoother or straighter, depending on how curved or straight each segment was originally.

- To smooth the curve of each selected fill outline or curved line, select the Selection tool and click the Smooth modifier in the Options section of the Tools panel, or select Modify > Shape > Smooth.
- To make small straightening adjustments on each selected fill outline or curved line, select the Selection tool and click the Straighten modifier in the Options section of the Tools panel, or select Modify > Shape > Straighten.
- To use shape recognition, select the Selection tool and click the Straighten modifier, or select Modify > Shape > Straighten.

See also

"Specify drawing preferences" on page 163

Optimize curves

Optimizing smooths curves by refining curved lines and filling outlines, reducing the number of curves used to define these elements. Optimizing curves also reduces the size of the Flash document (FLA file) and the exported Flash application (SWF file). Apply optimization to the same elements multiple times.

1 Select the drawn elements to optimize and select Modify > Shape > Optimize.

2 To specify the degree of smoothing, drag the Smoothing slider. The results depend on the curves selected. Generally, optimizing produces fewer curves, with less resemblance to the original outline.

3 Set additional options:

Use Multiple Passes Repeats the smoothing process until no further optimization can be accomplished; this is the same as repeatedly selecting Optimize with the same elements selected.

Show Totals Message Indicates the extent of the optimization when smoothing is complete.

4 Click OK.

Erase

Erasing with the Eraser tool removes strokes and fills.

Quickly delete everything on the Stage

❖ Double-click the Eraser tool .

Remove stroke segments or filled areas

1 Select the Eraser tool, and then click the Faucet modifier .

2 Click the stroke segment or filled area to delete.

Erase by dragging

1 Select the Eraser tool.

2 Click the Eraser Mode modifier and select an erasing mode:

Erase Normal Erases strokes and fills on the same layer.

Erase Fills Erases only fills; strokes are not affected.

Erase Lines Erases only strokes; fills are not affected.

Erase Selected Fills Erases only the currently selected fills and does not affect strokes, selected or not. (Select the fills to erase before using the Eraser tool in this mode.)

Erase Inside Erases only the fill on which you begin the eraser stroke. If you begin erasing from an empty point, nothing is erased. Strokes are unaffected by the eraser in this mode.

3 Click the Eraser Shape modifier and select an eraser shape and size. Make sure that the Faucet modifier is not selected.

4 Drag on the Stage.

Modify shapes

1 To convert lines to fills, select a line or multiple lines and select Modify > Shape > Convert Lines To Fills. Selected lines are converted to filled shapes, which allows you to fill lines with gradients or to erase a portion of a line. Converting lines to fills can make file sizes larger, but it can also speed up drawing for some animations.

2 To expand the shape of a filled object, select a filled shape, and select Modify > Shape > Expand Fill. Enter a value in pixels for Distance and select Expand or Inset For Direction. Expand enlarges the shape, and Inset reduces it.

This feature works best on a single, small, filled color shape with no stroke, that does not contain many small details.

3 To soften the edges of an object, select a filled shape, and select Modify > Shape > Soften Fill Edges. Set the following options:

Distance The width, in pixels, of the soft edge.

Number Of Steps Controls how many curves are used for the soft edge effect. The more steps you use, the smoother the effect. Increasing steps also creates larger files and slows drawing.

Expand Or Inset Controls whether the shape is enlarged or reduced to soften the edges.

This feature works best on a single filled shape that has no stroke, and can increase the file size of a Flash document and the resulting SWF file.

Snapping

About snapping

To automatically align elements with one another, use *snapping*. Flash provides three ways for you to align objects on the Stage:

- Object snapping snaps objects directly to other objects along their edges.
- Pixel snapping snaps objects directly to individual pixels or lines of pixels on the Stage.
- Snap alignment snaps objects to a specified *snap tolerance*, a preset boundary between objects and other objects or between objects and the edge of the Stage.

Note: *You can also snap to the grid or to guides.*

See also

"About the main toolbar and edit bar" on page 26

Use object snapping

To turn on object snapping, use the Snap To Objects modifier for the Selection tool, or the Snap To Objects command in the View menu.

If the Snap To Objects modifier for the Selection tool is on, a small black ring appears under the pointer when you drag an element. The small ring changes to a larger ring when the object is within snapping distance of another object.

See also

"Specify drawing preferences" on page 163

Turn object snapping on or off

❖ Select View > Snapping > Snap To Objects. A check mark appears next to the command when it is on.

When you move or reshape an object, the position of the Selection tool on the object provides the reference point for the snap ring. For example, if you move a filled shape by dragging near its center, the center point snaps to other objects. This is particularly useful for snapping shapes to motion paths for animating.

***Note:** For better control of object placement when snapping, begin dragging from a corner or center point.*

Adjust object snapping tolerances

1. Select Edit > Preferences (Windows) or Flash > Preferences (Macintosh), and click Editing.
2. Under Drawing Settings, adjust the Connect Lines setting.

Use pixel snapping

To turn on pixel snapping, use the Snap To Pixels command in the View menu. If Snap To Pixels is on, a pixel grid appears when the view magnification is set to 400% or higher. The pixel grid represents the individual pixels that appears in your Flash application. When you create or move an object, it is constrained to the pixel grid.

If you create a shape whose edges fall between pixel boundaries—for example, if you use a stroke with a fractional width, such as 3.5 pixels—Snap To Pixels snaps to pixel boundaries, and not to the edge of the shape.

- To turn pixel snapping on or off, select View > Snapping > Snap To Pixels. If the magnification is set to 400% or higher, a pixel grid is displayed. A check mark appears next to the command when it is on.
- To turn pixel snapping on or off temporarily, press the C key. When you release the C key, pixel snapping returns to the state you selected with View > Snapping > Snap To Pixels.
- To temporarily hide the pixel grid, press the X key. When you release the X key, the pixel grid reappears.

Use snap alignment

To turn on Snap Alignment, use the Snap Align command in the View menu. To select settings for Snap Alignment, use the Edit Snap Align command in the View menu.

When you select Snap Alignment settings, set the snap tolerance between horizontal or vertical edges of objects, and between objects' edges and the Stage border. You can also turn on snap alignment between the horizontal and the vertical centers of objects. All Snap Alignment settings are measured in pixels.

When Snap Alignment is turned on, dotted lines appear on the Stage when you drag an object to the specified snap tolerance. For example, if you set Horizontal snap tolerance to 18 pixels (the default setting), a dotted line appears along the edge of the object you are dragging when the object is exactly 18 pixels from another object. If you turn on Horizontal Center Alignment, a dotted line appears along the horizontal center vertices of two objects when you precisely align the vertices.

Select settings for Snap Alignment

1 Select View > Snapping > Edit Snap Align.

2 In the Snap Align dialog box, do any of the following:

- To set the snap tolerance between objects and the Stage border, enter a value for Movie Border.
- To set the snap tolerance between horizontal or vertical edges of objects, enter a value for Horizontal, Vertical, or both.
- To turn on Horizontal or Vertical Center Alignment, select Horizontal or Vertical Center Alignment or both.

Turn on Snap Alignment

❖ Select View > Snapping > Snap Align. A check mark appears next to the command when it is on.

Chapter 7: Working with color, strokes, and fills

Adobe® Flash® CS3 Professional includes many tools for creating your own artwork and for controlling the colors, strokes, and fills of your drawings. With Flash you can control and manipulate these aspects of your artwork both during and after its creation.

Working with color

About color

Adobe® Flash® CS3 Professional lets you apply, create, and modify colors. Using the default palette or a palette you create, you can choose colors to apply to the stroke or fill of an object you are about to create, or an object already on the Stage.

When applying a stroke color to a shape, you can do any of the following:

- Apply a solid color, gradient, or bitmap to a shape's fill. To apply a bitmap fill to a shape, you must import a bitmap into the current file. Select any solid color, gradient, and the style and weight of the stroke.
- Create an outlined shape with no fill by using No Color as a fill.
- Create a filled shape with no outline by using No Color as an outline.
- Apply a solid color fill to text.

With the Color panel, you can create and edit solid colors and gradient fills in RGB and HSB modes.

To access the system color picker, Alt-double-click (Windows) or Option-double-click (Macintosh) the Stroke Color or Fill Color control in the Tools panel, the shape Property inspector, or the Color panel.

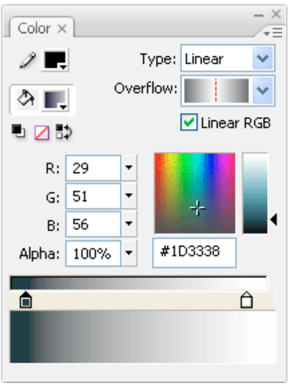
See also

"Setting text attributes" on page 271

About the Color panel

The Color panel lets you change the color of strokes and fills, including the following:

- Import, export, delete, and otherwise modify the color palette for a file by using the Swatches panel.
- Select colors in hexadecimal mode.
- Create multicolor gradients.
- Use gradients to produce a wide range of effects, such as giving an illusion of depth to a two-dimensional object.



Type: Linear

Overflow:

☑ Linear RGB

R: 29

G: 51

B: 56

Alpha: 100%

#1D3338

The Color panel with the gradient controls displayed.

Color panel options

Stroke Color Changes the color of the stroke, or the border, of a graphic object.

Fill Color Changes the color of the fill. The fill is the area of color that fills up the shape.

Type Menu Changes the fill style:

- **None** Removes the fill.
- **Solid** Provides a solid, single fill color.
- **Linear** Produces a gradient that blends on a linear path.
- **Radial** Produces a gradient that blends outward in a circular path from a central focal point.
- **Bitmap** Tiles the selected fill area with a bitmap image that you can select. When you choose Bitmap, a dialog box lets you select a bitmap image on your local computer, and add it to the library. You can apply this bitmap as a fill; the appearance is similar to a mosaic tile pattern with the image repeated within the shape.

RGB Lets you change the density of the red, green, and blue (RGB) colors in a fill.

Alpha Sets the opacity for a solid fill, or the currently selected slider for a gradient fill. An alpha value of 0% creates an invisible (or transparent) fill; an alpha value of 100% creates an opaque fill.

Current Color Swatch Displays the currently selected color. If you select a gradient fill type (Linear or Radial) from the fill Type menu, the Current Color Swatch displays the color transitions within the gradient you create.

System Color Picker Lets you select a color visually. Click System Color Picker and drag the cross-hair pointer until you find the color you want.

Hexadecimal value Displays the current color's hexadecimal value. To change the color using the hexadecimal value, type in a new value. Hexadecimal color values (also called hex values) are 6-digit alphanumeric combinations that represent a color.

Overflow Lets you control colors applied past the limits of a linear or radial gradient.

- **Extend** (Default) Applies the colors you specify past the end of the gradient.
- **Reflect** Causes the gradient colors to fill the shape using a reflective mirroring effect. The gradients you specify are repeated in a pattern from the beginning of the gradient to the end, and then repeated in the opposite sequence from the end of the gradient to the beginning, and then back to the beginning of the gradient to the end until the selected shape is filled.
- **Repeat** Repeats the gradient from the beginning of the gradient to the end until the selected shape is filled.

***Note:** Overflow modes are supported only in Flash Player 8 and later.*

Linear RGB Creates a Scalable Vector Graphics (SVG)-compliant linear or radial gradient.

Modifying color palettes

About color palettes

Each Flash file contains its own color palette, stored in the Flash document. Flash displays a file's palette as swatches in the Fill Color and Stroke Color controls and in the Swatches panel. The default color palette is the web-safe palette of 216 colors. To add colors to the current color palette, use the Color panel.

You can import and export both solid and gradient color palettes between Flash files, as well as between Flash and other applications.

See also

"Work with solid colors and gradient fills in the Color panel" on page 188

Duplicate, delete, and clear colors

Duplicate colors in the palette, delete individual colors, or clear all colors from the palette.

- To duplicate or delete a color, select Window > Swatches, click the color to duplicate or delete, and select Duplicate Swatch or Delete Swatch from the panel menu. When duplicating a swatch, the paint bucket appears. Click in the empty area of the Swatches panel with the paint bucket to make a duplicate of the selected color.
- To clear all colors from the color palette, in the Swatches panel, select Clear Colors from the panel menu. All colors except black and white are removed from the palette.

Use the default palette and the web-safe palette

Save the current palette as the default palette, replace the current palette with the default palette defined for the file, or load the web-safe palette to replace the current palette.

- To load or save the default palette, in the Swatches panel, select one of the following commands from the menu in the upper-right corner:

Load Default Colors Replaces the current palette with the default palette.

Save As Default Saves the current color palette as the default palette. The new default palette is used when you create new files.

- To load the web-safe 216-color palette, in the Swatches panel, select Web 216 from the menu in the upper-right corner

Sort color by hue in the palette

To make it easier to locate a color, sort colors in the palette by hue.

❖ In the Swatches panel, select Sort by Color from the menu in the upper-right corner.

Import and export color palettes

To import and export both RGB colors and gradients between Flash files, use Flash Color Set files (CLR files). Import and export RGB color palettes by using Color Table files (ACT files). You can also import color palettes, but not gradients, from GIF files. You cannot import or export gradients from ACT files.

Import a color palette

1 In the Swatches panel, select one of the following commands from the menu in the upper-right corner:

- To append the imported colors to the current palette, select Add Colors.
- To replace the current palette with the imported colors, select Replace Colors.

2 Navigate to the desired file, select it, and click OK.

Export a color palette

1 In the Swatches panel, select Save Colors from the menu in the upper-right corner and enter a name for the color palette.

2 For Save As Type (Windows) or Format (Macintosh), select Flash Color Set or Color Table. Click Save.

Strokes, fills, and gradients

Use the Stroke Color and Fill Color controls in the Tools panel

The Tools panel Stroke Color and Fill Color controls set the painting attributes of new objects you create with the drawing and painting tools. To use these controls to change the painting attributes of existing objects, first select the objects on the Stage.

***Note:** Gradient swatches appear only in the Fill Color control.*

- Click the triangle next to the Stroke or Fill color control, and select a color swatch. You can select gradients only for the fill color.
- Click the System Color Picker button in the pop-up window, and select a color.
- Type a color’s hexadecimal value in the box.
- To return to the default color settings (white fill and black stroke), click the Black And White button in the Tools panel.
- To remove any stroke or fill, click the No Color button.

***Note:** The No Color button appears only when you are creating an oval or rectangle. You can create an object without a stroke or fill, but you cannot use the No Color button with an existing object. Instead, select the existing stroke or fill and delete it.*

- To Swap colors between the fill and the stroke, click the Swap Colors button in the Tools panel.

Use the Stroke Color and Fill Color controls in the Property inspector

To change the stroke color, style, and weight for a selected object, use the Stroke Color control in the Property inspector. For stroke style, choose from styles that are preloaded with Flash, or create a custom style. To select a solid color fill, use the Fill Color control in the Property inspector.

Select a stroke color, style, and weight by using the Property inspector

1 Select an object or objects on the Stage (for symbols, first double-click to enter symbol-editing mode).

2 Select Window > Properties > Properties.

3 To select a stroke style, click the triangle next to the Style menu and select an option from the menu. To create a custom style, select Custom from the Property inspector, select options in the Stroke Style dialog box, and click OK.

***Note:** Selecting a stroke style other than Solid can increase file size.*

4 To select a stroke weight, click the triangle next to the Weight menu and set the slider.

5 Specify a stroke height by doing one of the following:

- Select one of the preset values from the Height menu. Preset values appear in points.
- Type a value from 0 to 200 in the height text field, and press Enter.

6 To enable stroke hinting, select the Stroke Hinting check box. Stroke hinting adjusts line and curve anchors on full pixels, preventing blurry vertical or horizontal lines.

7 To set the style for a path end, select a Cap option:

None Is flush with the path's end

Round Adds a round cap that extends beyond the path end by half the stroke width

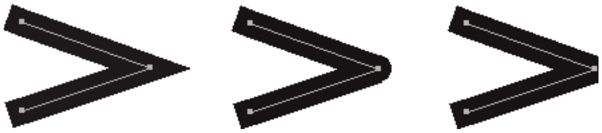
Square Adds a square cap that extends beyond the path by half the stroke width

8 (Optional) If you are drawing lines using the Pencil or Brush tools with the drawing mode set to Smooth, use the Smoothing slider to specify the degree to which Flash smooths the lines you draw.

By default, the Smoothing value is set to 50, but you can specify a value from 0 to 100. The greater the smoothing value, the smoother the resulting line.

***Note:** When the drawing mode is set to Straighten or Ink, the Smoothing slider is disabled.*

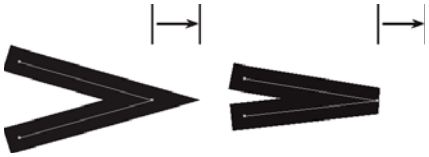
9 To define how two path segments meet, select a Join option. To change the corners in an open or closed path, select a path and select another join option.



Miter, round, and bevel joins

10 To avoid beveling a Miter join, enter a Miter limit.

Line lengths exceeding this value are squared instead of pointed. For example, a Miter limit of 2 for a 3-point stroke means that when the length of the point is twice the stroke weight, Flash removes the limit point.



Applying a Miter limit

Apply a solid color fill by using the Property inspector

1 Select a closed object or objects on the Stage.

2 Select Window > Properties > Properties.

3 To select a color, click the triangle next to the Fill color control and do one of the following:

- Select a color swatch from the palette.
- Type a color's hexadecimal value in the box.

Creating gradients

A gradient is a multicolor fill in which one color gradually changes into another color. Flash lets you apply up to 15 color transitions to a gradient. Flash can create two types of gradients:

Linear gradients change color along a single axis (horizontal or vertical).

Radial gradients change color in an outward direction starting from a central focal point. You can adjust the direction of a gradient, its colors, the location of the focal point, and many other properties of the gradient.

Adobe® Flash® CS3 Professional provide additional control over linear and radial gradients for use with Flash Player. These controls, called overflow modes, let you specify how colors are applied beyond the gradient.

For a sample of gradients, see the Flash Samples page at www.adobe.com/go/learn_fl_samples. Download and decompress the Samples zip file and navigate to the Graphics\AnimationAndGradients folder to access the sample.

Work with solid colors and gradient fills in the Color panel

You can create any color using the Color panel. If an object is selected on the Stage, the color modifications you make in the Color panel are applied to the selection. You can select colors in RGB or HSB, or you can expand the panel to use hexadecimal mode. You can also specify an alpha value to define the degree of transparency for a color. In addition, you can select a color from the existing color palette.

You can expand the Color panel to display a larger color space in place of the color bar, a split color swatch showing the current and previous colors, and a Brightness slider to modify color brightness in all color modes.

Create or edit a solid color with the Color panel

1 To apply the color to existing artwork, select an object or objects on the Stage, and select Window > Color.

2 To select a color mode display, select RGB (the default setting) or HSB from the panel menu in the upper-right corner.

3 Click the Stroke or Fill icon to specify which attribute to modify.

***Note:** Click the icon, not the color control, or the Color Picker opens.*

4 If you selected the Fill icon in step 3, verify that Solid is selected in the Type menu.

5 If an object is selected on the Stage, the color modifications you make in the Color panel are applied to the selection. Do one of the following:

- To select a color, click in the color space in the Color panel. To adjust the brightness of the color, drag the Brightness slider.

***Note:** To create colors other than black or white, make sure the Brightness slider is not set to either extreme.*

- Enter values in the color value boxes: Red, Green, and Blue values for RGB display; Hue, Saturation, and Brightness values for HSB display; or hexadecimal values for hexadecimal display. Enter an Alpha value to specify the degree of transparency, from 0 for complete transparency to 100 for complete opacity.

- To return to the default color settings, black and white (white fill and black stroke), click the Default Stroke and Fill button.
- To swap colors between the fill and the stroke, click the Swap Stroke and Fill button.
- To apply no color to the fill or stroke, click the No Color button.

***Note:** You cannot apply a stroke or fill of No Color to an existing object. Instead, select the existing stroke or fill, and delete it.*

- Click the Stroke or Fill color control, and select a color.

6 To add the color defined in step 7 to the color swatch list for the current document, select Add Swatch from the menu in the upper-right corner.

Create or edit a gradient fill with the Color panel

1 To apply a gradient fill to existing artwork, select an object or objects on the Stage.

2 If the Color panel is not visible, select Window > Color.

3 To select a color mode display, select RGB (the default setting) or HSB.

4 Select a gradient type from the Type menu:

Linear Creates a gradient that shades from the starting point to the end point in a straight line.

Radial Produces a gradient that blends outward in a circular path from a central focal point.

***Note:** When you select a linear or radial gradient, the Color panel also includes two other options if you are publishing for Flash Player 8. First, the Overflow menu appears below the Type menu. Use the Overflow menu to control the colors applied past the limits of the gradient. Second, the gradient definition bar appears, with pointers below the bar indicating the colors in the gradient.*

5 (Optional) To apply to the gradient, select an overflow mode: Extend (the default mode), Reflect, and Repeat.

6 (Optional) To create an SVG-compliant (Scalable Vector Graphics) linear or radial gradient, select the Linear RGB check box.

7 To change a color in the gradient, select one of the color pointers below the gradient definition bar and double-click the color space that appears directly below the gradient bar to display the Color Picker. Drag the Brightness slider to adjust the lightness of the color.

8 To add a pointer to the gradient, click on or below the gradient definition bar. Select a color for the new pointer, as described in step 7.

You can add up to 15 color pointers, letting you create a gradient with up to 15 color transitions.

9 To reposition a pointer on the gradient, drag the pointer along the gradient definition bar. Drag a pointer down and off of the gradient definition bar to remove it.

10 To save the gradient, click the triangle in the upper-right corner of the Color panel, and select Add Swatch from the menu.

The gradient is added to the Swatches panel for the current document.

Modify strokes with the Ink Bottle tool

To change the stroke color, width, and style of lines or shape outlines, use the Ink Bottle tool. You can apply only solid colors, not gradients or bitmaps, to lines or shape outlines.

Using the Ink Bottle tool, rather than selecting individual lines, makes it easier to change the stroke attributes of multiple objects at one time.

1 Select the Ink Bottle tool from the Tools panel.

2 Select a stroke color.

3 Select a stroke style and stroke width from the Property inspector.

4 To apply the stroke modifications, click an object on the Stage.

Apply solid, gradient, and bitmap fills with the Paint Bucket tool

The Paint Bucket tool fills enclosed areas with color. This tool lets you do the following:

- Fill empty areas, and change the color of already painted areas.
- Paint with solid colors, gradient fills, and bitmap fills.
- Use the Paint Bucket tool to fill areas that are not entirely enclosed.
- Have Flash close gaps in shape outlines as you use the Paint Bucket tool.

1 Select the Paint Bucket tool from the Tools panel.

2 Select a fill color and style.

3 Click the Gap Size modifier and select a gap size option:

- Don’t Close Gaps to close gaps manually before filling the shape. Closing gaps manually can be faster for complex drawings.
- A Close option to have Flash fill a shape that has gaps.

***Note:** If gaps are too large, you might have to close them manually.*

4 Click the shape or enclosed area to fill.

See also

“Use the Stroke Color and Fill Color controls in the Property inspector” on page 186

“Working with imported bitmaps” on page 155

Transform gradient and bitmap fills

You can transform a gradient or bitmap fill by adjusting the size, direction, or center of the fill.

1 Select the Gradient Transform tool from the Tools panel.

2 Click an area filled with a gradient or bitmap fill. A bounding box with editing handles appears. When the pointer is over any one of these handles, it changes to indicate the function of the handle.

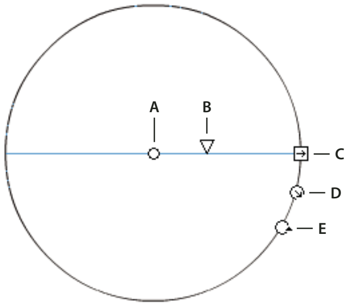
Center point The rollover icon for the center point handle is a four-way arrow.

Focal point The focal point handle appears only when you select a radial gradient. The rollover icon for the focal point handle is an inverted triangle.

Size The rollover icon for the size handle (middle handle icon on the edge of the bounding box) is a circle with an arrow inside of it.

Rotation Adjusts the rotation of the gradient. The rollover icon for the rotation handle (the bottom handle icon on the edge of the bounding box) is four arrows in the shape of a circle.

Width Adjusts the width of the gradient. The rollover icon for the width handle (the square handle) is a double-ended arrow.

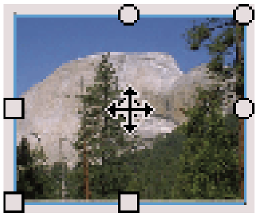


Radial Gradient controls
A. *Center point* ***B.*** *Width* ***C.*** *Rotation* ***D.*** *Size* ***E.*** *Focal point*

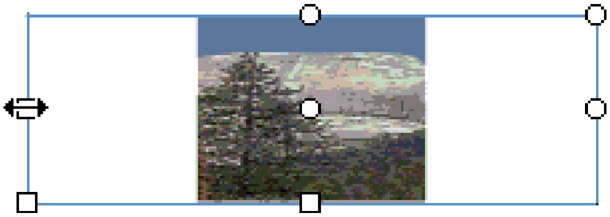
Press Shift to constrain the direction of a linear gradient fill to multiples of 45˚.

3 Reshape the gradient or fill in any of the following ways:

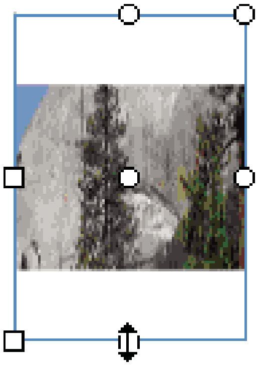
- To reposition the center point of the gradient or bitmap fill, drag the center point.



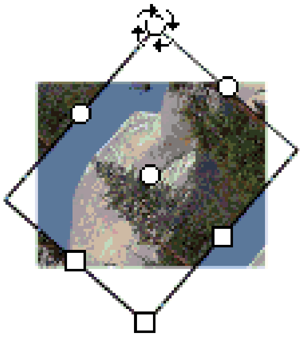
- To change the width of the gradient or bitmap fill, drag the square handle on the side of the bounding box. (This option resizes only the fill, not the object containing the fill.)



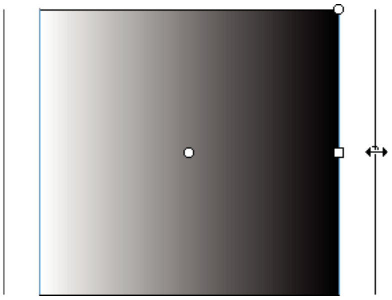
- To change the height of the gradient or bitmap fill, drag the square handle at the bottom of the bounding box.



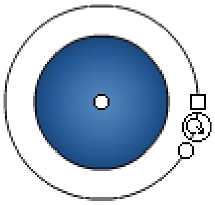
- To rotate the gradient or bitmap fill, drag the circular rotation handle at the corner. You can also drag the lowest handle on the bounding circle of a circular gradient or fill.



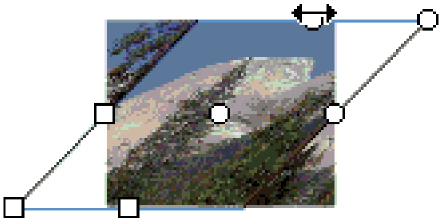
- To scale a linear gradient or a fill, drag the square handle at the center of the bounding box.



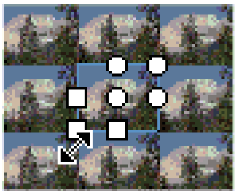
- To change the focal point of a circular gradient, drag the middle circular handle on the bounding circle.



- To skew or slant a fill within a shape, drag one of the circular handles on the top or right side of the bounding box.



- To tile a bitmap inside a shape, scale the fill.



***Note:** To see all the handles when working with large fills or fills close to the edge of the Stage, select View > Pasteboard.*

Copy strokes and fills with the Eyedropper tool

Use the Eyedropper tool to copy fill and stroke attributes from one object and immediately apply them to another object. The Eyedropper tool also lets you sample the image in a bitmap to use as a fill.

1 To apply the attributes of a stroke or filled area to another stroke or filled area, select the Eyedropper tool and click the stroke or filled area whose attributes you want to apply.

When you click a stroke, the tool automatically changes to the Ink Bottle tool. When you click a filled area, the tool automatically changes to the Paint Bucket tool with the Lock Fill modifier turned on.

2 Click another stroke or filled area to apply the new attributes.

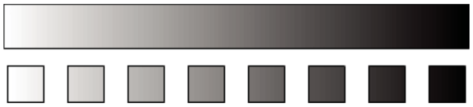
See also

"Break apart groups and objects" on page 202

Lock a gradient or bitmap to fill the Stage

You can lock a gradient or bitmap fill to make it appear that the fill extends over the entire Stage and that the objects painted with the fill are masks revealing the underlying gradient or bitmap.

When you select the Lock Fill modifier with the Brush or Paint Bucket tool and paint with the tool, the bitmap or gradient fill extends across the objects you paint on the Stage.



Using the Lock Fill modifier creates the appearance of a single gradient or bitmap fill being applied to separate objects on the Stage.

See also

"Apply solid, gradient, and bitmap fills with the Paint Bucket tool" on page 190

Use a locked gradient fill

1 Select the Brush or Paint Bucket tool and select a gradient or bitmap as a fill.

2 Select Linear or Radial from the Type menu in the Color panel.

3 Click the Lock Fill modifier 🔒.

4 First paint the areas where you want to place the center of the fill, and then move to other areas.

Use a locked bitmap fill

1 Select the bitmap to use.

2 Select Bitmap from the Type menu in the Color panel.

3 Select the Brush or Paint Bucket tool.

4 Click the Lock Fill modifier 🔒.

5 First paint the areas where you want to place the center of the fill, and then move to other areas.

Chapter 8: Working with graphic objects

In Adobe® Flash® CS3 Professional, you can work with several different kinds of graphic objects. Each kind has its own advantages and disadvantages. By understanding the capabilities of the different object types, you can make good decisions about which types of objects to use in your work.

About graphic objects

Understanding graphic objects in Flash

In Flash, graphic objects are items on the Stage. Flash lets you move, copy, delete, transform, stack, align, and group graphic objects.

Modifying lines and shapes can alter other lines and shapes on the same layer.

***Note:** "Graphic objects" in Flash are different from "ActionScript objects," which are part of the ActionScript™ programming language. Do not confuse the two uses of the term "objects." For more information on objects in the programming language, see About data types in Learning ActionScript 2.0 in Adobe Flash, or Data types in Programming ActionScript 3.0.*

See also

"Drawing" on page 159

About shapes

Shapes are one type of graphic object you can create in Flash. When you draw shapes that overlap each other in the same layer, the topmost shape cuts away the part of the shape underneath it that it overlaps. In this way, drawing shapes is a destructive drawing mode.

When a shape has both a stroke and a fill, these are considered separate graphic elements, which can be selected and moved independently.

Depending on the type of drawing you are doing, you may find that this behavior well suited to the final art you want to create.

To draw shapes

1 Deselect the Object Drawing option in the Tools panel.

2 Select a drawing tool, and draw on the Stage.

Do not select the Rectangle Primitive or Oval Primitive tools, as these tools create shape primitives instead of standard shapes.

About drawing objects

Drawing objects are graphic elements you create with the Flash drawing tools in object drawing mode. When a tool is in object drawing mode, the shapes you create with it are self-contained. The stroke and fill of a shape are not separate elements, and shapes that overlap do not alter one another.

Depending on your drawing style and the content you want to create, object drawing mode may be best suited to your goals.

To enter object drawing mode

1 Select the Object Drawing option in the Tools panel.

2 Select a drawing tool from the Tools panel and draw on the Stage.

About primitive objects

Primitive objects are graphic shapes that allow you to adjust their characteristics in the Property inspector. This lets you precisely control the size, corner radius, and other properties of the shape at any time after you have created it without having to re-draw it from scratch.

Two types of primitives are available, rectangles and ovals.

To draw a primitive object

1 Select the Rectangle Primitive Tool or Oval Primitive Tool from the Tools panel.

2 Draw on the Stage.

Selecting objects

Selecting objects

To modify an object, select it first. You can group individual objects to manipulate them as a single object. Modifying lines and shapes can alter other lines and shapes on the same layer. When you select objects or strokes, Flash highlights them with a marquee.

You can choose to select only an object’s strokes or only its fills. You can hide selection highlighting to edit objects without viewing highlighting.

When you select an object, the Property inspector displays the following:

- The object’s stroke and fill, its pixel dimensions, and the *x* and *y* coordinates of the object’s transformation point
- A mixed selection, if you select multiple items. The pixel dimensions and x and y coordinates of the selected set of items.

You can use a shape’s Property inspector to change that object’s stroke and fill.

To prevent a group or symbol from being selected and accidentally changed, lock the group or symbol.

See also

“Drawing” on page 159

“Working with color, strokes, and fills” on page 183

“Group objects” on page 201

“Symbols overview” on page 207

Select objects with the Selection tool

The Selection tool lets you select entire objects by clicking an object or dragging to enclose the object within a rectangular selection marquee.

***Note:** To select the Selection tool, you can also press the V key. To temporarily switch to the Selection tool when another tool is active, hold down the Control key (Windows) or Command key (Macintosh).*

To disable the Shift-selecting option, deselect the option in Flash General Preferences. See "Set preferences in Flash" on page 27. Instances, groups, and type blocks must be completely enclosed to be selected.

- To select a stroke, fill, group, instance, or text block, click the object.
- To select connected lines, double-click one of the lines.
- To select a filled shape and its stroked outline, double-click the fill.
- To select objects within a rectangular area, drag a marquee around the object or objects to select.
- To add to a selection, hold down the Shift key while making additional selections.
- To select everything on every layer of a scene, select Edit > Select All, or press Control+A (Windows) or Command+A (Macintosh). Select All doesn't select objects on locked or hidden layers, or layers not on the current Timeline.
- To deselect everything on every layer, select Edit > Deselect All, or press Control+Shift+A (Windows) or Command+Shift+A (Macintosh).
- To select everything on one layer between keyframes, click a frame in the Timeline.
- To lock or unlock a group or symbol, select the group or symbol, and then select Modify > Arrange > Lock. Select Modify > Arrange > Unlock All to unlock all locked groups and symbols.

Select objects with the Lasso tool

When you use the Lasso tool and its Polygon Mode modifier, you can switch between the freehand and straight-edged selection modes.

Draw a freehand selection area

1 Drag the Lasso tool around the area.

2 End the loop approximately where you started, or let Flash automatically close the loop with a straight line.

Draw a straight-edged selection area

1 Select the Lasso tool's Polygon Mode modifier in the options area of the Tools panel.

2 Click to set the starting point.

3 Position the pointer where you want the first line to end, and click. Continue setting end points for additional line segments.

4 To close the selection area, double-click.

Draw a selection area with both freehand and straight-line edges

1 Deselect the Polygon Mode modifier of the Lasso tool.

2 To draw a freehand segment, drag the Lasso tool on the Stage.

3 To draw straight-edged segments, hold down the Alt key (Windows) or Option key (Macintosh) and click to set start and end points for each new line segment.

4 To close the selection area, do one of the following:

- Release the mouse button; Flash will close the selection area for you.
- Double-click on the starting end of the selection area line.

Hide selection highlighting

Hiding highlighting while you are selecting and editing objects lets you see how artwork appears in its final state.

❖ Select View > Hide Edges.

Select the command again to show selection highlighting.

Set custom bounding box colors for selected objects

You can set different colors to be used for the bounding box rectangles that appear around different kinds of selected objects on the Stage.

1 Select Edit > Preferences.

2 Click the General category.

3 In the Highlight Color section, select a color for each type of object and click OK.

Moving, copying, and deleting objects

Moving and copying objects

When you move an object, the Property inspector indicates the new position.

Copy an object by dragging it, pasting it, or by using the Transform panel.

Move objects

To move an object, you can drag the object, use the arrow keys, use the Property inspector, or use the Info panel.

Move objects by dragging

1 Select an object or multiple objects.

2 Select the Selection tool ↖, position the pointer over the object, and do one of the following:

- To move the object, drag it to the new position.
- To copy the object and move the copy, Alt-drag (Windows) or Option-drag (Macintosh).
- To constrain the object’s movement to multiples of 45˚, Shift-drag.

Move objects by using the arrow keys

1 Select an object or multiple objects.

2 Do one of the following:

- To move the selection 1 pixel at a time, press the arrow key for the direction in which you want the object to move.
- To move the selection 10 pixels at a time, press Shift+arrow key.

***Note:** When Snap To Pixels is selected, the arrow keys move objects by pixel increments on the document's pixel grid, not by pixels on the screen.*

Move objects by using the Property inspector

1 Select an object or multiple objects.

2 If the Property inspector is not visible, select Window > Properties > Properties.

3 Enter *x* and *y* values for the location of the upper-left corner of the selection.

The units are relative to the upper-left corner of the Stage.

***Note:** The Property inspector uses the units specified for the Ruler Units option in the Document Properties dialog box.*

Move objects by using the Info panel

1 Select an object or multiple objects.

2 If the Info Panel is not visible, select Window > Info.

3 Enter *x* and *y* values for the location of the upper-left corner of the selection.

The units are relative to the upper left corner of the Stage.

Move and copy objects by pasting

To move or copy objects between layers, scenes, or other Flash files, use the pasting technique. You can paste an object in a position relative to its original position.

1 Select an object or multiple objects.

2 Select Edit > Cut or Edit > Copy.

3 Select another layer, scene, or file, and then select Edit > Paste In Place to paste the selection in the same position relative to the Stage. Select Edit > Paste In Center to paste the selection in the center of the work area.

Copying objects with the clipboard

Elements copied to the clipboard are anti-aliased, so they look as good in other applications as they do in Flash. This feature is particularly useful for frames that include a bitmap image, gradients, transparency, or a mask layer.

Graphics pasted from other Flash documents or programs are placed in the current frame of the current layer. How a graphic element is pasted into a Flash scene depends on the type of element it is, its source, and the preferences you have set:

- Text from a text editor becomes a single text object.
- Vector-based graphics from any drawing program become a group that can be ungrouped and edited.
- Bitmaps become a single grouped object just like imported bitmaps. You can break apart pasted bitmaps or convert pasted bitmaps to vector graphics.

***Note:** Before pasting graphics from Illustrator into Flash, convert colors to RGB in Illustrator.*

Copy transformed objects

You can create a scaled, rotated, or skewed copy of an object.

1 Select an object.

2 Select Window > Transform.

3 Enter scale, rotation, or skew values.

4 Click the Create Copy button in the Transform panel.

Delete objects

Deleting an object removes it from the file. Deleting an instance of an object on the Stage does not delete the symbol from the library.

1 Select an object or multiple objects.

2 Do one of the following:

- Press Delete or Backspace.
- Select Edit > Clear.
- Select Edit > Cut.
- Right-click (Windows) or Control-click (Macintosh) the object, and select Cut from the context menu.

See also

“Scale objects” on page 205

“Skew objects” on page 206

“Rotate objects” on page 205

“Create or open a document and set its properties” on page 52

“Convert bitmaps to vector graphics” on page 158

“Use pixel snapping” on page 181

Arranging objects

Stack objects

In a layer, Flash stacks objects in the order in which they are created, placing the most recently created object at the top of the stack. The stacking order of objects determines how they appear when they overlap. You can change the stacking order of objects at any time.

Drawn lines and shapes always appear below groups and symbols on the stack. To move them up the stack, you must group them or make them into symbols.

Layers also affect the stacking order. Everything on Layer 2 appears in front of everything on Layer 1, and so on. To change the order of layers, drag the layer name in the Timeline to a new position.

1 Select the object.

2 Do one of the following:

- Select Modify > Arrange > Bring To Front Or Send To Back to move the object or group to the top or bottom of the stacking order.
- Select Modify > Arrange > Bring Forward Or Send Backward to move the object or group forward or backward one position in the stacking order.

If more than one group is selected, the groups move in front of or behind all unselected groups, while maintaining their order relative to each other.

See also

"About layers" on page 36

Align objects

The Align panel lets you align selected objects along the horizontal or vertical axis. You can align objects vertically along the right edge, center, or left edge of the selected objects, or horizontally along the top edge, center, or bottom edge of the selected objects.

For a tutorial about the layout tools in Flash, see Use Layout Tools on the Flash Tutorials page at www.adobe.com/go/learn_fl_tutorials.

1 Select the objects to align.

2 Select Window > Align.

3 To apply alignment modifications relative to Stage dimensions, in the Align panel, select To Stage.

4 To modify the selected object(s), select alignment buttons.

Group objects

To manipulate elements as a single object, group them. For example, after creating a drawing, you might group the elements of the drawing so that you can easily select and move the drawing as a whole.

When you select a group, the Property inspector displays the *x* and *y* coordinates of the group and its pixel dimensions.

You can edit groups without ungrouping them. You can also select an individual object in a group for editing without ungrouping the objects.

Group and ungroup objects

❖ Select the objects to group. You can select shapes, other groups, symbols, text, and so on.

- To group objects, select Modify > Group, or press Control+G (Windows) or Command+G (Macintosh).
- To ungroup objects, select Modify > Ungroup, or press Control+Shift+G (Windows) or Command+Shift+G (Macintosh).

Edit a group or an object within a group

1 Select the group, and then select Edit > Edit Selected, or double-click the group with the Selection tool.

Everything on the page that is not part of the group is dimmed, indicating that elements outside the group are inaccessible.

2 Edit any element within the group.

3 Select Edit > Edit All, or double-click a blank spot on the Stage with the Selection tool.

Flash restores the group to its status as a single entity, and you can work with other elements on the Stage.

Break apart groups and objects

To separate groups, instances, and bitmaps into ungrouped, editable elements, you break them apart, which significantly reduces the file size of imported graphics.

Although you can select Edit > Undo immediately after breaking apart a group or object, breaking apart is not entirely reversible. It affects objects as follows:

- Severs a symbol instance's link to its master symbol
- Discards all but the current frame in an animated symbol
- Converts a bitmap to a fill
- Places each character into a separate text block when applied to text blocks
- Converts characters to outlines when applied to a single text character.

Do not confuse the Break Apart command with the Ungroup command. The Ungroup command separates grouped objects, returning grouped elements to the state they were in before grouping. It does not break apart bitmaps, instances, or type, or convert type to outlines.

1 Select the group, bitmap, or symbol to break apart.

2 Select Modify > Break Apart.

***Note:** Breaking apart animated symbols, or groups in an interpolated animation is not recommended and might have unpredictable results. Breaking apart complex symbols and large blocks of text can take a long time. You might need to increase the application's memory allocation to properly break apart complex objects.*

See also

"Break text apart" on page 270

Transforming objects

Transforming objects

You can transform graphic objects, as well as groups, text blocks, and instances, by using the Free Transform tool or the options in the Modify > Transform menu. Depending on the type of element you select, you can transform, rotate, skew, scale, or distort the element. You can change or add to a selection during a transformation operation.

When you transform an object, group, text box, or instance, the Property inspector for that item displays any changes made to the item's dimensions or position.

A bounding box appears during transform operations that involve dragging. The bounding box is rectangular (unless it was modified with the Distort command or the Envelope modifier) with its edges initially aligned parallel to the edges of the Stage. Transformation handles are located on each corner and in the middle of each side. As you drag, the bounding box previews the transformations.

Move, realign, change, and track the transformation point

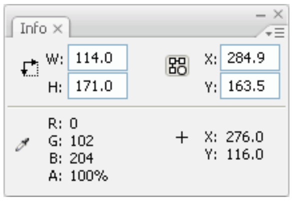
During a transformation, a transformation point appears at the center of a selected element. The transformation point is initially aligned with the object's center point. You can move the transformation point, return it to its default location, and move the default point of origin.

For scaling, skewing, or rotating graphic objects, groups, and text blocks, the point opposite the point you drag is the point of origin by default. For instances, the transformation point is the point of origin by default. You can move the default point of origin for a transformation.

You begin a transformation by selecting the Free Transform tool , or selecting one of the Modify > Transform commands. Once you have begun a transformation, you can track the location of the transformation point in the Info panel and in the Property inspector.

- To move the transformation point, drag it from within the selected graphic object.
- To realign the transformation point with the element's center point, double-click the transformation point.
- To switch the point of origin for a scale or skew transformation, hold down the Alt key (Windows) or Option key (Macintosh) while dragging your chosen object control point during the transformation.
- To display the transformation point coordinates in the Info panel, click the Registration/Transformation Point button in the Info panel. The lower-right square in the button becomes a circle to indicate the registration point coordinates are being displayed.

When you select the center square, the X and Y values to the right of the coordinate grid in the Info panel display the *x* and *y* coordinates of the transformation point. In addition, the X and Y values for the transformation point appear in the Property inspector for the symbol.



W: 114.0
H: 171.0
X: 284.9
Y: 163.5
R: 0
G: 102
B: 204
A: 100%
X: 276.0
Y: 116.0

Coordinate grid; Info panel with Registration/Transformation Point button in transformation mode, and with x and y coordinates of selection transformation point visible

By default, the Registration/Transformation Point button is in registration mode, and the X and Y values display the location of the upper-left corner of the current selection, relative to the upper-left corner of the Stage.

***Note:** For symbol instances, the X and Y values display the location of the symbol registration point, or the location of the upper-left corner of the symbol instance.*

Transform objects freely

You can perform individual transformations or combine several transformations, such as moving, rotating, scaling, skewing, and distortion.

***Note:** The Free Transform tool cannot transform symbols, bitmaps, video objects, sounds, gradients, or text. If a multiple selection contains any of these items, only the shape objects are distorted. To transform a text block, first convert the characters to shape objects.*

1 Select a graphic object, group, instance, or text block on the Stage.

2 Click the Free Transform tool .

Moving the pointer over and around the selection changes the pointer to indicate which transformation function is available.

3 To transform the selection, drag the handles:

- To move the selection, position the pointer over the object within the bounding box, and drag the object to a new position. Do not drag the transformation point.
- To set the center of rotation or scaling, drag the transformation point to a new location.
- To rotate the selection, position the pointer just outside a corner handle and drag. The selection rotates around the transformation point. Shift-drag to rotate in 45˚ increments.
- To rotate around the opposite corner, Alt-drag (Windows) or Option-drag (Macintosh).
- To scale the selection, drag a corner handle diagonally to scale in two dimensions. Shift-drag to resize proportionally.
- To scale in the respective direction only, drag a corner handle or a side handle horizontally or vertically.
- To skew the selection, position the pointer on the outline between the transformation handles and drag.
- To distort shapes, press Control (Windows) or Command (Macintosh) and drag a corner handle or a side handle.
- To *taper* the object—to move the selected corner and the adjoining corner equal distances from their origins, shift-Control-click to drag (Windows) or Shift-Command-click to drag (Macintosh) a corner handle.

4 To end the transformation, click outside the selected item.

Distort objects

When you apply a Distort transformation to a selected object, dragging a corner handle or an edge handle on the bounding box moves the corner or edge and realigns the adjoining edges. Shift-drag a corner point to constrain the distortion to a taper—that is, move that corner and the adjoining corner an equal distance and in the opposite direction from each other. The adjoining corner is the corner on the same axis as the direction you drag. Control-click (Windows) or Command-click (Macintosh) to drag a middle point on an edge to move the entire edge freely.

You can distort graphic objects by using the Distort command. You can also distort objects when performing a free transform on them.

***Note:** The Distort command cannot modify symbols, shape primitives, bitmaps, video objects, sounds, gradients, object groups, or text. If a multiple selection contains any of these items, only the shape objects are distorted. To modify text, first convert the characters to shape objects.*

1 Select a graphic object or objects on the Stage.

2 Select Modify > Transform > Distort.

3 Place the pointer on one of the transformation handles and drag.

4 To end the transformation, click outside the selected object or objects.

Modify shapes with the Envelope modifier

The Envelope modifier lets you warp and distort objects. An envelope is a bounding box that contains one or more objects. Changes made to an envelope's shape affect the shape of the objects in the envelope. You edit the shape of an envelope by adjusting its points and tangent handles.

***Note:** The Envelope modifier cannot modify symbols, bitmaps, video objects, sounds, gradients, object groups, or text. If a multiple selection contains any of these items, only the shape objects are distorted. To modify text, first convert the characters to shape objects.*

1 Select a shape on the Stage.

2 Select Modify > Transform > Envelope.

3 Drag the points and tangent handles to modify the envelope.

Scale objects

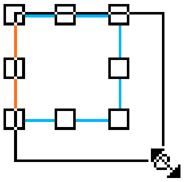
Scaling an object enlarges or reduces the object horizontally, vertically, or both.

1 Select a graphic object or objects on the Stage.

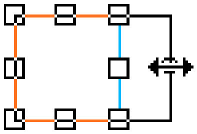
2 Select Modify > Transform > Scale.

3 Do one of the following:

- To scale the object both horizontally and vertically, drag one of the corner handles. Proportions are maintained as you scale. Shift-drag to scale nonuniformly.



- To scale the object either horizontally or vertically, drag a center handle.



4 To end the transformation, click outside the selected object or objects.

Note: *When you increase the size of a number of items, items near the edges of the bounding box might be moved off the Stage. If this occurs, select View > Pasteboard to see the elements that are beyond the edges of the Stage.*

See also

"About 9-slice scaling and movie clip symbols" on page 222

"Edit movie clip symbols with 9-slice scaling" on page 223

Rotate objects

Rotating an object turns it around its transformation point. The transformation point is aligned with the registration point, which defaults to the center of the object, but you can move the point by dragging it.

You can rotate an object with the following methods when using the Rotate commands:

- Dragging with the Free Transform tool (you can skew and scale the object in the same operation).
- By specifying an angle in the Transform panel (you can scale the object in the same operation).

Rotate and skew objects by dragging

1 Select the object or objects on the Stage.

2 Select Modify > Transform > Rotate And Skew.

3 Do one of the following:

- Drag a corner handle to rotate the object.
- Drag a center handle to skew the object.

4 To end the transformation, click outside the selected object or objects.

Rotate objects by 90°

1 Select the object or objects.

2 Select Modify > Transform > Rotate 90° CW to rotate clockwise, or Rotate 90° CCW to rotate counterclockwise.

Skew objects

Skewing an object transforms it by slanting it along one or both axes. You can skew an object by dragging or by entering a value in the Transform panel.

1 Select the object or objects.

2 Select Window > Transform.

3 Click Skew.

4 Enter angles for the horizontal and vertical values.

Flip objects

You can flip objects across their vertical or horizontal axis without moving their relative position on the Stage.

1 Select the object.

2 Select Modify > Transform > Flip Vertical or Flip Horizontal.

Restore transformed objects

When you use the Transform panel to scale, rotate, and skew instances, groups, and fonts, Flash saves the original size and rotation values with the object. This process lets you remove the transformations you applied and restore the original values.

You can undo only the most recent transformation performed in the Transform panel when you select Edit > Undo. You can reset all transformations performed in the Transform panel by clicking the Reset button in the panel before you deselect the object.

Restore a transformed object to its original state

1 Select the transformed object.

2 Select Modify > Transform > Remove Transform.

Reset a transformation performed in the Transform panel

❖ With the transformed object still selected, click the Reset button in the Transform panel.

Chapter 9: Using symbols, instances, and library assets

Adobe® Flash® CS3 Professional allows you to import and create many kinds of assets to populate your Flash documents. These assets are managed in Flash as *symbols*, *instances*, and *library assets*. Understanding how the types of assets work together lets you make good choices about how and when to use them, and anticipate the best design options for your work.

Working with symbols

Symbols overview

A *symbol* is a graphic, button, or movie clip that you create once in the Flash authoring environment or by using the Button (AS 2.0), SimpleButton (AS 3.0), and MovieClip classes. You can then reuse the symbol throughout your document or in other documents.

A symbol can include artwork that you import from another application. Any symbol that you create automatically becomes part of the library for the current document.

An *instance* is a copy of a symbol located on the Stage or nested inside another symbol. An instance can be different from its symbol in color, size, and function. Editing the symbol updates all of its instances, but applying effects to an instance of a symbol updates only that instance.

Using symbols in your documents dramatically reduces file size; saving several instances of a symbol requires less storage space than saving multiple copies of the contents of the symbol. For example, you can reduce the file size of your documents by converting static graphics, such as background images, into symbols and then reusing them. Using symbols can also speed SWF file playback, because a symbol needs to be downloaded to Flash Player only once.

Share symbols among documents as shared library assets during authoring or at runtime. For runtime shared assets, you can link assets in a source document to any number of destination documents, without importing the assets into the destination document. For assets shared during authoring, you can update or replace a symbol with any other symbol available on your local network.

If you import library assets with the same name as assets already in the library, you can resolve naming conflicts without accidentally overwriting existing assets.

For a video tutorial about using symbols and instances, see www.adobe.com/go/vid0122.

For a text tutorial about symbols and instances, see Create Symbols and Instances on the Flash Tutorials page at www.adobe.com/go/learn_fl_tutorials.

See also

“Working with button symbols” on page 220

“Using shared library assets” on page 218

“Managing media assets with the Flash document library” on page 64

Types of symbols

Each symbol has a unique Timeline and Stage, complete with layers. You can add frames, keyframes, and layers to a symbol Timeline, just as you can to the main Timeline. When you create a symbol you choose the symbol type.

- Use graphic symbols for static images and to create reusable pieces of animation that are tied to the main Timeline. Graphic symbols operate in sync with the main Timeline. Interactive controls and sounds won't work in a graphic symbol's animation sequence. Graphic symbols add less to the FLA file size than buttons or movie clips because they have no timeline.
- Use button symbols to create interactive buttons that respond to mouse clicks, rollovers, or other actions. You define the graphics associated with various button states, and then assign actions to a button instance. For more information, see Handling events in *Learning ActionScript 2.0 in Adobe Flash* or Handling events in *Programming ActionScript 3.0*.
- Use movie clip symbols to create reusable pieces of animation. Movie clips have their own multiframe Timeline that is independent from the main Timeline—think of them as nested inside a main Timeline that can contain interactive controls, sounds, and even other movie clip instances. You can also place movie clip instances inside the Timeline of a button symbol to create animated buttons. In addition, movie clips are scriptable with ActionScript™.
- Use font symbols to export a font and use it in other Flash documents.

Flash provides built-in *components*, movie clips with defined parameters, that you can use to add user interface elements, such as buttons, check boxes, or scroll bars, to your documents. For more information, see About components in *Using ActionScript 2.0 Components*, or About ActionScript 3.0 components in *Using ActionScript 3.0 Components*.

Note: *To preview animation in component instances and scaling of 9-slice-scaled movie clips in the Flash authoring environment, select Control > Enable Live Preview.*

See also

"Embed and share fonts" on page 268

Create symbols

You can create a symbol from selected objects on the Stage, create an empty symbol and make or import the content in symbol-editing mode, and create font symbols in Flash. Symbols can contain all the functionality that Flash, including animation.

Using symbols that contain animation lets you create Flash applications with a lot of movement while minimizing file size. Consider creating animation in a symbol that has a repetitive or cyclic action—the up-and-down motion of a bird's wings, for example.

To add symbols to your document, use shared library assets during authoring or at runtime.

A tutorial entitled Create Symbols and Instances is available on the Flash Tutorials web page at www.adobe.com/go/learn_fl_tutorials. Download and decompress the Tutorials zip file and navigate to the Basic Tasks\Create Symbols and Instances directory.

See also

“About changing instance properties” on page 213

“Using shared library assets” on page 218

“Embed and share fonts” on page 268

Convert selected elements to a symbol

1 Select an element or several elements on the Stage. Do one of the following:

- Select Modify > Convert To Symbol.
- Drag the selection to the Library panel.
- Right-click (Windows) or Control-click (Macintosh) and select Convert To Symbol from the context menu.

2 In the Convert To Symbol dialog box, type the name of the symbol and select the behavior.

3 Click in the registration grid to position the registration point for the symbol.

4 Click OK.

Flash adds the symbol to the library. The selection on the Stage becomes an instance of the symbol. Once you have created a symbol, you can edit it in symbol edit mode by choosing Edit > Edit Symbols, or you can edit it in the context of the Stage by choosing Edit > Edit In Place. You can also change the registration point of a symbol.

Create an empty symbol

1 Do one of the following:

- Select Insert > New Symbol.
- Click the New Symbol button at the lower left of the Library panel.
- Select New Symbol from the Library Panel menu in the upper-right corner of the Library panel.

2 In the Create New Symbol dialog box, type the name of the symbol and select the behavior.

3 Click OK.

Flash adds the symbol to the library and switches to symbol-editing mode. In symbol-editing mode, the name of the symbol appears above the upper left corner of the Stage, and a cross hair indicates the symbol’s registration point.

4 To create the symbol content, use the Timeline, draw with the drawing tools, import media, or create instances of other symbols.

5 To return to document-editing mode, do one of the following:

- Click the Back button.
- Select Edit > Edit Document.
- Click the scene name in the Edit bar.

When you create a symbol, the registration point is placed at the center of the window in symbol-editing mode. You can place the symbol contents in the window in relation to the registration point. To change the registration point, when you edit a symbol, move the symbol contents in relation to the registration point.

Convert animation on the Stage into a movie clip

To reuse an animated sequence on the Stage, or to manipulate it as an instance, select it and save it as a movie clip symbol.

1 On the main Timeline, select every frame in every layer of the animation on the Stage that you want to use. For information on selecting frames, see “Manage frames and keyframes in the Timeline” on page 69.

2 Do one of the following to copy the frames:

- Right-click (Windows) or Control-click (Macintosh) any selected frame, and select Copy Frames from the context menu. To delete the sequence after converting it to a movie clip, select Cut.
- Select Edit > Timeline > Copy Frames. To delete the sequence after converting it to a movie clip, select Cut Frames.

3 Deselect your selection and make sure nothing on the Stage is selected. Select Insert > New Symbol.

4 Name the symbol. For Type, select Movie Clip, then click OK.

5 On the Timeline, click Frame 1 on Layer 1, and select Edit > Timeline > Paste Frames.

This action pastes the frames (and any layers and layer names) you copied from the main Timeline to the Timeline of this movie clip symbol. Any animation, buttons, or interactivity from the frames you copied now becomes an independent animation (a movie clip symbol) that you can reuse.

6 To return to document-editing mode, do one of the following:

- Click the Back button.
- Select Edit > Edit Document.
- Click the scene name in the Edit bar above the Stage.

Duplicate symbols

Duplicating a symbol lets you use an existing symbol as a starting point for creating a symbol.

To create versions of the symbol with different appearances, also use instances.

Duplicate a symbol using the Library panel

❖ Select a symbol in the Library panel and do one of the following:

- Right-click (Windows) or Control-click (Macintosh), and select Duplicate from the context menu.
- Select Duplicate from the Library Panel menu.

Duplicate a symbol by selecting an instance

1 Select an instance of the symbol on the Stage.

2 Select Modify > Symbol > Duplicate Symbol.

The symbol is duplicated, and the instance is replaced with an instance of the duplicate symbol.

Edit symbols

When you edit a symbol, Flash updates all the instances of that symbol in your document. Edit the symbol in the following ways:

- In context with the other objects on the Stage by using the Edit In Place command. Other objects are dimmed to distinguish them from the symbol you are editing. The name of the symbol you are editing appears in an Edit bar at the top of the Stage, to the right of the current scene name.

- In a separate window, using the Edit In New Window command. Editing a symbol in a separate window lets you see the symbol and the main Timeline at the same time. The name of the symbol you are editing appears in the Edit bar at the top of the Stage.

You edit the symbol by changing the window from the Stage view to a view of only the symbol, using symbol-editing mode. The name of the symbol you are editing appears in the Edit bar at the top of the Stage, to the right of the current scene name.

When you edit a symbol, Flash updates all instances of the symbol throughout the document to reflect your edits. While editing a symbol, use any of the drawing tools, import media, or create instances of other symbols.

- Change the registration point of a symbol (the point identified by the coordinates 0, 0) by using any symbol-editing method.

Edit a symbol in place

1 Do one of the following:

- Double-click an instance of the symbol on the Stage.
- Select an instance of the symbol on the Stage and right-click (Windows) or Control-click (Macintosh), and select Edit in Place.
- Select an instance of the symbol on the Stage, and select Edit > Edit In Place.

2 Edit the symbol.

3 To change the registration point, drag the symbol on the Stage. A cross hair indicates the location of the registration point.

4 To exit edit-in-place mode and return to document-editing mode, do one of the following:

- Click the Back button.
- Select the current scene name from the Scene menu in the Edit bar.
- Select Edit > Edit Document.
- Double-click outside the symbol content.

Edit a symbol in a new window:

1 Select an instance of the symbol on the Stage and right-click (Windows) or Control-click (Macintosh), and select Edit In New Window.

2 Edit the symbol.

3 To change the registration point, drag the symbol on the Stage. A cross hair indicates the location of the registration point.

4 Click the Close box in the upper-right corner (Windows) or upper-left corner (Macintosh) to close the new window, and click in the main document window to return to editing the main document.

Edit a symbol in symbol-editing mode

1 Do one of the following to select the symbol:

- Double-click the symbol’s icon in the Library panel.
- Select an instance of the symbol on the Stage, and right-click (Windows) or Control-click (Macintosh), and select Edit from the context menu.
- Select an instance of the symbol on the Stage and select Edit > Edit Symbols.

- Select the symbol in the Library panel and select Edit from the Library Panel menu, or right-click (Windows) or Control-click (Macintosh) the symbol in the Library panel and select Edit.

2 Edit the symbol.

3 To exit symbol-editing mode and return to editing the document, do one of the following:

- Click the Back button at the left of the Edit bar at the top of the Stage.
- Select Edit > Edit Document.
- Click the scene name in the Edit bar at the top of the Stage.
- Double-click outside the symbol content.

Working with symbol instances

Create instances

After you create a symbol, you can create instances of that symbol throughout your document, including inside other symbols. When you modify the symbol, Flash updates all instances of the symbol.

You can give names to instances from the Property inspector. Use the instance name to refer to an instance in ActionScript. To control instances with ActionScript, give each instance within a single timeline a unique name. For more information, see Handling events in *Learning ActionScript 2.0 in Adobe Flash* or Handling events in *Programming ActionScript 3.0*.

To specify color effects, assign actions, set the graphic display mode, or change the behavior of new instances, use the Property inspector. The behavior of the instance is the same as the symbol behavior, unless you specify otherwise. Any changes you make affect only the instance and not the symbol.

For a video tutorial about using symbols and instances, see www.adobe.com/go/vid0122.

For a text tutorial about symbols and instances, see Create Symbols and Instances on the Flash Tutorials page at www.adobe.com/go/learn_fl_tutorials.

Create an instance of a symbol

1 Select a layer in the Timeline. Flash can place instances only in keyframes, always on the current layer. If you don't select a keyframe, Flash adds the instance to the first keyframe to the left of the current frame.

***Note:** A keyframe is a frame in which you define a change in the animation. For more information, see "Manage frames and keyframes in the Timeline" on page 69.*

2 Select Window > Library.

3 Drag the symbol from the library to the Stage.

4 If you created an instance of a graphic symbol, to add the number of frames that will contain the graphic symbol, select Insert > Timeline > Frame.

Apply a custom name to an instance

1 Select the instance on the Stage.

2 Select Window > Properties > Properties, and enter a name in the Instance Name box.

About changing instance properties

Each symbol instance has its own properties that are separate from the symbol. You can change the tint, transparency, and brightness of an instance; redefine how the instance behaves (for example, change a graphic to a movie clip); and specify how animation plays inside a graphic instance. You can also skew, rotate, or scale an instance without affecting the symbol.

In addition, you can name a movie clip or button instance so that you can use ActionScript to change its properties. For more information, see Classes in *Learning ActionScript 2.0 in Adobe Flash* or Objects and classes in *Programming ActionScript 3.0*. To edit instance properties, you use the Property inspector (Windows > Properties > Properties).

The properties of an instance are saved with it. If you edit a symbol or relink an instance to a different symbol, any instance properties you've changed still apply to the instance.

Change the color and transparency of an instance

Each instance of a symbol can have its own color effect. To set color and transparency options for instances, use the Property inspector. Settings in the Property inspector also affect bitmaps placed in symbols.

When you change the color and transparency for an instance in a specific frame, Flash makes the change as soon as it displays that frame. To make gradual color changes, apply a motion tween. When tweening color, you enter different effect settings in starting and ending keyframes of an instance, and then tween the settings to make the instance's colors shift over time.

***Note:** If you apply a color effect to a movie clip symbol that has multiple frames, Flash applies the effect to every frame in the movie clip symbol.*

1 Select the instance on the Stage, and select Window > Properties > Properties.

2 In the Property inspector, select one of the following options from the Color menu:

Brightness Adjusts the relative lightness or darkness of the image, measured on a scale from black (–100%) to white (100%). To adjust brightness, click the triangle and drag the slider or enter a value in the box.

Tint Colors the instance with the same hue. To set the tint percentage from transparent (0%) to completely saturated (100%), use the Tint slider in the Property inspector. To adjust tint, click the triangle and drag the slider or enter a value in the box. To select a color, enter red, green, and blue values in the respective boxes, or click the color control and select a color from the Color Picker.

Alpha Adjusts the transparency of the instance, from transparent (0%) to completely saturated (100%). To adjust the alpha value, click the triangle and drag the slider or enter a value in the box.

Advanced Separately adjusts the red, green, blue, and transparency values of an instance. This is most useful to create and animate subtle color effects on objects such as bitmaps. The controls on the left let you reduce the color or transparency values by a specified percentage. The controls on the right let you reduce or increase the color or transparency values by a constant value.

The current red, green, blue, and alpha values are multiplied by the percentage values, and then added to the constant values in the right column, producing the new color values. For example, if the current red value is 100, setting the left slider to 50% and the right slider to 100% produces a new red value of 150 ([100 x .5] + 100 = 150).

***Note:** The Advanced settings in the Effect panel implement the function (a * y+ b)= x where a is the percentage specified in the left set of boxes, y is the color of the original bitmap, b is the value specified in the right set of boxes, and x is the resulting effect (between 0 and 255 for RGB, and 0 and 100 for alpha transparency).*

You can also change the color of an instance using the ActionScript ColorTransform object. For detailed information on the Color object, see ColorTransform in *ActionScript 2.0 Language Reference* or *ActionScript 3.0 Language and Components Reference*.

See also

"Add a motion tween to instances, groups, or type" on page 241

Swap one instance for another

To display a different instance on the Stage and preserve all the original instance properties, such as color effects or button actions, assign a different symbol to an instance.

For example, suppose you're creating a cartoon with a rat symbol for your character, but decide to change the character to a cat. You could replace the rat symbol with the cat symbol and have the updated character appear in roughly the same location in all your frames.

See also

"Working with button symbols" on page 220

Assign a different symbol to an instance

1 Select the instance on the Stage, and select Window > Properties > Properties.

2 Click the Swap button in the Property inspector.

3 Select a symbol to replace the symbol currently assigned to the instance. To duplicate a selected symbol, click Duplicate Symbol and click OK.

Duplicating lets you base a new symbol on an existing one in the library and minimizes copying if you're making several symbols that differ slightly.

Replace all instances of a symbol

❖ Drag a symbol with the same name as the symbol you are replacing from one Library panel into the Library panel of the FLA file you are editing and click Replace. If you have folders in the Library, the new symbol must be dragged into the same folder as the symbol you are replacing.

Change an instance's type

To redefine an instance's behavior in a Flash application, change its type. For example, if a graphic instance contains animation that you want to play independently of the main Timeline, redefine the graphic instance as a movie clip instance.

1 Select the instance on the Stage, and select Window > Properties > Properties.

2 Select Graphic, Button, or Movie Clip from the menu in the Property inspector.

Set looping for graphic instances

To determine how animation sequences inside a graphic instance play in your Flash application, set options in the Property inspector.

An animated graphic symbol is tied to the Timeline of the document in which the symbol is placed. In contrast, a movie clip symbol has its own independent Timeline. Animated graphic symbols, because they use the same Timeline as the main document, display their animation in document-editing mode. Movie clip symbols appear as static objects on the Stage and do not appear as animations in the Flash editing environment.

1 Select a graphic instance on the Stage, and select Window > Properties > Properties.

2 Select an animation option from the menu below the instance name:

Loop Loops all the animation sequences contained in the current instance for as many frames as the instance occupies.

Play Once Plays the animation sequence beginning from the frame you specify to the end of the animation and then stops.

Single Frame Displays one frame of the animation sequence. Specify which frame to display.

Break apart an instance symbol

To break the link between an instance and a symbol and make the instance into a collection of ungrouped shapes and lines, you *break apart* the instance. This feature is useful for changing the instance substantially without affecting any other instance. If you modify the source symbol after breaking apart the instance, the instance is not updated with the changes.

1 Select the instance on the Stage.

2 Select Modify > Break Apart. This action breaks the instance into its component graphic elements.

3 To modify these elements, use the painting and drawing tools.

Get information about instances on the Stage

The Property inspector and Info panel display the following information about instances selected on the Stage:

- In the Property inspector, view the instance's behavior and settings—for all instance types, color effect settings, location, and size; for graphics, the loop mode and first frame that contains the graphic; for buttons, the instance name (if assigned) and tracking option; for movie clips, the instance name (if assigned). For location, the Property inspector displays the *x* and *y* coordinates of either the symbol's registration point or the symbol's upper left corner, depending on which option is selected in the Info panel.
- In the Info panel, view the instance's size and location; the location of its registration point; its red (R), green (G), blue (B), and alpha (A) values (if the instance has a solid fill); and the location of the pointer. The Info panel also displays the *x* and *y* coordinates of either the symbol's registration point or the symbol's upper-left corner, depending on which option is selected. To display the coordinates of the registration point, click the center square in the Coordinate grid in the Info panel. To display the coordinates of the upper-left corner, click the upper-left square in the Coordinate grid.
- In the Movie Explorer, view the contents of the current document, including instances and symbols.

View any actions assigned to a button or movie clip in the Actions panel.

See also

"Using the Movie Explorer with screens" on page 370

Get information about an instance

1 Select the instance on the Stage.

2 Display the Property inspector (Window > Properties > Properties) or panel to use:

- To display the Info panel, select Window > Info.
- To display the Movie Explorer, select Window > Movie Explorer.
- To display the Actions panel, select Window > Actions.

View the symbol definition for the selected symbol in the Movie Explorer

1 Click the Show Buttons, Movie Clips, and Graphics button at the top of the Movie Explorer.

2 Right-click (Windows) or Control-click (Macintosh), and select Show Symbol Instances and Go To Symbol Definition; or select these options from the menu in the upper-right corner of the Movie Explorer.

Jump to the scene containing instances of a selected symbol

1 Display the symbol definitions.

2 Right-click (Windows) or Control-click (Macintosh), and select Show Movie Elements and Go To Symbol Definition; or select these options from the menu in the upper-right corner of the Movie Explorer.

Library assets

Copy library assets between documents

Copy library assets from a source document into a destination document in a variety of ways: by copying and pasting the asset, by dragging and dropping the asset, or by opening the library of the source document in the destination document and dragging the source document assets into the destination document.

Share symbols between documents as shared library assets during authoring or at runtime.

If you attempt to copy assets that have the same name as existing assets in the destination document, the Resolve Library Conflicts dialog box lets you choose whether to overwrite the existing assets or to preserve the existing assets and add the new assets with modified names. Organize library assets in folders to minimize name conflicts when copying assets between documents.

See also

“Work with folders in the Library panel” on page 66

“Working with button symbols” on page 220

“Using shared library assets” on page 218

Copy a library asset by copying and pasting

1 Select the asset on the Stage in the source document.

2 Select Edit > Copy.

3 Make the destination document the active document.

4 To paste the asset in the center of the visible pasteboard, place the pointer on the Stage and select Edit > Paste In Center. To place the asset in the same location as in the source document, select Edit > Paste In Place.

Copy a library asset by dragging

❖ With the destination document open, select the asset in the Library panel in the source document and drag the asset into the Library panel in the destination document.

Copy a library asset by opening the source document library in the destination document

1 With the destination document active, select File > Import > Open External Library.

2 Select the source document, and click Open.

3 Drag an asset from the source document library onto the Stage or into the library of the destination document.

Conflicts between library assets

If you import or copy a library asset into a document that already contains a different asset of the same name, choose whether to replace the existing item with the new item. This option is available with all the methods for importing or copying library assets.

The Resolve Library Items dialog box appears when you attempt to place items that conflict with existing items in a document. A conflict exists when you copy an item from a source document that already exists in the destination document and the items have different modification dates. Avoid naming conflicts by organizing your assets inside folders in your document's library. The dialog box also appears when you paste a symbol or component into your document's Stage and you already have a copy of the symbol or component that has a different modification date from the one you're pasting.

If you choose not to replace the existing items, Flash attempts to use the existing item instead of the conflicting item that you are pasting. For example, if you copy a symbol named Symbol 1 and paste the copy into the Stage of a document that already contains a symbol named Symbol 1, Flash creates an instance of the existing Symbol 1.

If you choose to replace the existing items, Flash replaces the existing items (and all their instances) with the new items of the same name. If you cancel the Import or Copy operation, the operation is canceled for all items (not just those items that conflict in the destination document).

Only identical library item types may be replaced with each other. That is, you cannot replace a sound named Test with a bitmap named Test. In such cases, the new items are added to the library with the word Copy appended to the name.

Note: *Replacing library items using this method is not irreversible. Save a backup of your FLA file before you perform complex paste operations that are resolved by replacing conflicting library items.*

If the Resolve Library Conflict dialog box appears when you are importing or copying library assets into a document, resolve the naming conflict.

Resolve naming conflicts between library assets

❖ Do one of the following:

- To preserve the existing assets in the destination document, click Don't Replace Existing Items.
- To replace the existing assets and their instances with the new items of the same name, click Replace Existing Items.

Using shared library assets

About shared library assets

Shared library assets let you use assets from a source document in multiple destination documents:

- For runtime shared assets, assets from a source document are linked as external files in a destination document. Runtime assets are loaded into the destination document during document playback—that is, at runtime. The source document containing the shared asset does not need to be available on your local network when you author the destination document. The source document must be posted to a URL for the shared asset to be available to the destination document at runtime.
- For shared assets during authoring, update or replace any symbol in a document you are authoring with any other symbol available on your local network. Update the symbol in the destination document as you author the document. The symbol in the destination document retains its original name and properties, but its contents are updated or replaced with those of the symbol you select.

Using shared library assets can optimize workflow and document asset management.

Working with runtime shared assets

Using runtime shared library assets involves two procedures. First, the author of the source document defines a shared asset in the source document and enters an identifier string for the asset and a URL *(HTTP or HTTPS only)* where the source document will be posted.

Second, the author of the destination document defines a shared asset in the destination document and enters an identifier string and URL identical to those used for the shared asset in the source document. Alternatively, the destination document author can drag the shared assets from the posted source document into the destination document library. The ActionScript version set in the Publish settings must match that of the source document.

In either scenario, the source document must be posted to the specified URL for the shared assets to be available for the destination document.

Define runtime shared assets in a source document

To define sharing properties for an asset in a source document, and make the asset accessible for linking to destination documents, use the Symbol Properties dialog box or the Linkage Properties dialog box.

1 With the source document open, select Window > Library:

- Select a movie clip, button, or graphic symbol in the Library panel, and select Properties from the Library Panel menu. Click Advanced.
- Select a font symbol, sound, or bitmap, and select Linkage from the Library Panel menu.

2 For Linkage, select Export For Runtime Sharing to make the asset available for linking to the destination document.

3 Enter an identifier for the symbol. Do not include spaces. This is the name Flash uses to identify the asset when linking to the destination document.

Note: *Flash also uses the linkage identifier to identify a movie clip or button that is used as an object in ActionScript. See Working with movie clips in Learning ActionScript 2.0 in Adobe Flash or Working with movie clips in Programming ActionScript 3.0.*

4 Enter the URL where the SWF file containing the shared asset will be posted, and click OK.

When you publish the SWF file, you must post the SWF file to the URL you specified, so that the shared assets are available to destination documents.

Link to runtime shared assets from a destination document

Link a shared asset to a destination document by entering the identifier and URL

1 In the destination document, select Window > Library.

- Select a movie clip, button, graphic symbol, bitmap, or sound in the Library panel, and select Properties from the Library Panel menu. Click Advanced.
- Select a font symbol, and select Linkage from the Library Panel menu.

2 For Linkage, select Import For Runtime Sharing to link to the asset in the source document.

3 Enter an identifier for the symbol, bitmap, or sound that is identical to the identifier used for the symbol in the source document. Do not include spaces.

4 Enter the URL where the SWF source file containing the shared asset is posted, and click OK.

Link a shared asset to a destination document by dragging

1 In the destination document, do one of the following:

- Select File > Open.
- Select File > Import > Open External Library.

2 Select the source document and click Open.

3 Drag the shared asset from the source document Library panel into the Library panel or onto the Stage in the destination document.

Turn off sharing for a symbol in a destination document

1 In the destination document, select the linked symbol in the Library panel and do one of the following:

- If the asset is a movie clip, button, or graphic symbol, select Properties from the Library Panel menu.
- If the asset is a font symbol, select Linkage from the Library Panel menu.

2 Deselect Import For Runtime Sharing, and click OK.

Update or replace symbols

You can update or replace a movie clip, button, or graphic symbol in a document with any other symbol in a FLA file accessible on your local network. The original name and properties of the symbol in the destination document are preserved, but the contents of the symbol are replaced with the contents of the symbol you select. Any assets that the selected symbol uses are also copied into the destination document.

1 With the document open, select a movie clip, button, or graphic symbol and select Properties from the Library Panel menu.

2 If the Linkage and Source areas of the Symbol Properties dialog box are not showing, click Advanced.

3 To select a new FLA file, click Browse.

4 Navigate to a FLA file that contains the symbol to use to update or replace the selected symbol in the Library panel, and click Open.

5 Navigate to a symbol, and click OK.

6 In the Symbol Properties dialog box, under Source, select Always Update Before Publishing and click OK.

Working with button symbols

Create a button

Buttons are actually four-frame interactive movie clips. When you select the button behavior for a symbol, Flash creates a Timeline with four frames. The first three frames display the button's three possible states; the fourth frame defines the active area of the button. The Timeline doesn't actually play; it reacts to pointer movement and actions by jumping to the appropriate frame.

To make a button interactive, you place an instance of the button symbol on the Stage and assign actions to the instance. You must assign the actions to the instance of the button in the document, not to frames in the button's Timeline.

Each frame in the Timeline of a button symbol has a specific function:

- The first frame is the Up state, representing the button whenever the pointer is not over the button.
- The second frame is the Over state, representing the button's appearance when the pointer is over the button.
- The third frame is the Down state, representing the button's appearance as it is clicked.
- The fourth frame is the Hit state, defining the area that responds to the mouse click. This area is invisible in the SWF file.

Create a button using a movie clip symbol or a button component. Using each type of button has advantages. Creating a button using a movie clip lets you add more frames to the button or add more complex animation. However, movie clip buttons have a larger file size than button symbols. Using a button component allows you to bind the button to other components, to share and display data in an application. Button components also include prebuilt features, such as accessibility support, and can be customized. Button components include the Button, RadioButton, and CheckBox. For more information, see Button component in *ActionScript 2.0 Components Language Reference* or Using the Button in *Using ActionScript 3.0 Components*.

For a text tutorial about buttons, see Add Button Navigation and Animation on the Flash Tutorials page at www.adobe.com/go/learn_fl_tutorials.

1 Select Edit > Deselect All to ensure that nothing is selected on the Stage.

2 Select Insert > New Symbol, or press Control+F8 (Windows) or Command+F8 (Macintosh).

To create the button, you convert the button frames to keyframes.

3 In the Create New Symbol dialog box, enter a name for the new button symbol, and for Type select Button.

Flash switches to symbol-editing mode. The Timeline header changes to display four consecutive frames labeled Up, Over, Down, and Hit. The first frame, Up, is a blank keyframe.

4 To create the Up state button image, select the Up frame in the Timeline and then use the drawing tools, import a graphic, or place an instance of another symbol on the Stage.

You can use a graphic or movie clip symbol in a button, but you cannot use another button in a button. Use a movie clip symbol to animate the button.

5 Click the Over frame, and select Timeline > Keyframe.

Flash inserts a keyframe that duplicates the contents of the Up frame.

6 Change or edit the button image for the Over state.

7 Repeat steps 5 and 6 for the Down frame and the Hit frames.

The Hit frame is not visible on the Stage, but it defines the area of the button that responds when clicked. The graphic for the Hit frame must be a solid area large enough to encompass all the graphic elements of the Up, Down, and Over frames. It can also be larger than the visible button. If you do not specify a Hit frame, the image for the Up state is used as the Hit frame.

To create a disjoint rollover, in which moving the pointer over a button causes another graphic on the Stage to change, place the Hit frame in a different location than the other button frames.

8 To assign a sound to a state of the button, select that state's frame in the Timeline, select Window > Properties > Properties, and then select a sound from the Sound menu in the Property inspector.

9 When you finish, select Edit > Edit Document. To create an instance of the button in the document, drag the button symbol from the Library panel.

See also

"Using sounds in Flash" on page 291

Enable, edit, and test buttons

By default, Flash keeps buttons disabled as you create them, to make it easier to select and work with them. When a button is disabled, clicking the button selects it. When a button is enabled, it responds to the mouse events that you've specified as if the SWF file were playing. You can still select enabled buttons. Disable buttons as you work, and enable buttons to quickly test their behavior.

For a text tutorial about buttons, see Add Button Navigation and Animation on the Flash Tutorials page at www.adobe.com/go/learn_fl_tutorials.

Enable and disable buttons

❖ Select Control > Enable Simple Buttons. A check mark appears next to the command to indicate buttons are enabled. Select the command again to disable buttons.

Any buttons on the Stage now respond. As you move the pointer over a button, Flash displays the Over frame; when you click within the button's active area, Flash displays the Down frame.

Select, move, or edit an enabled button

❖ Do one of the following:

- Use the Selection tool to drag a selection rectangle around the button.
- Use the arrow keys to move the button.
- If the Property inspector is not visible, select Window > Properties > Properties to edit the button in the Property inspector, or Alt+double-click (Windows) or Option+double-click the button (Macintosh).

Test a button

❖ Do one of the following:

- Select Control > Enable Simple Buttons. Move the pointer over the enabled button to test it.
- Select the button in the Library panel, and click the Play button in the Library preview window.

- Select Control > Test Scene or Control > Test Movie.

Movie clips in buttons are not visible in the Flash authoring environment.

Scaling and caching symbols

About 9-slice scaling and movie clip symbols

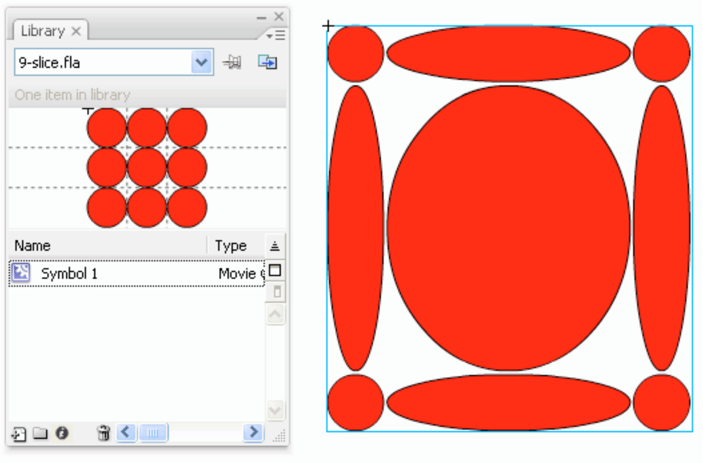
9-slice scaling allows you to specify how scaling is applied to specific areas of a movie clip. With 9-slice scaling, you can ensure that the movie clip looks correct when scaled. With normal scaling, Flash scales all parts of a movie clip equally, and in both the horizontal and vertical dimension. For many movie clips, this equal scaling can make the clip's graphics look strange, especially at the corners of rectangular movie clips. This is often true of movie clips used as user interface elements, such as buttons.

The movie clip is visually divided into nine sections with a grid-like overlay, and each of the nine areas is scaled independently. To maintain the visual integrity of the movie clip, corners are not scaled, while the remaining areas of the image are scaled (as opposed to being stretched) larger or smaller, as needed.

When a movie clip symbol has 9-slice scaling applied, it appears in the Library panel preview with the guides displayed. If Enable Live Preview is turned on (Control > Enable Live Preview) when you scale instances of the movie clip on the Stage, you see the 9-slice scaling applied on the Stage.

***Note:** 9-slice scaling cannot be applied to Graphic or Button symbols. Bitmaps inside 9-slice enabled movie clips are scaled normally, without 9-slice distortion, while the other movie clip contents are scaled according to the 9-slice guides.*

A 9-slice-enabled movie clip can contain nested objects within it, but only certain types of objects inside the movie clip properly scale in the slice-9 manner. To make a movie clip with internal objects that also adhere to 9-slice scaling when the movie clip is scaled, those nested objects must be shapes, drawing objects, groups, or graphic symbols.



A 9-slice-enabled symbol in the Library and scaled on the Stage.

For video tutorials about 9-slice scaling, see:

- www.adobe.com/go/vid0204
- www.adobe.com/go/vid0205

Edit movie clip symbols with 9-slice scaling

By default, slice guides are placed at 25% (or one-fourth) of the symbol's width and height from the edge of the symbol. In symbol-editing mode, the slice guides appear as dotted lines superimposed on the symbol. The slice guides don't snap when you drag them on the pasteboard. The guides do not appear when the symbol is on the Stage.

You cannot edit 9-slice-enabled symbols in place on the Stage. You must edit them in symbol-editing mode.

***Note:** Instances made from a 9-slice-enabled movie clip symbol can be transformed, but should not be edited. Editing these instances can have unpredictable results.*

For video tutorials about 9-slice scaling, see:

- www.adobe.com/go/vid0204
- www.adobe.com/go/vid0205

Enable 9-slice scaling for an existing movie clip symbol

1 With the source document open, select Window > Library.

2 Select a movie clip, button, or graphic symbol in the Library panel.

3 Select Properties from the Library Panel menu.

4 Select Enable Guides for 9-slice Scaling.

Edit a 9-slice-enabled movie clip symbol

1 Enter symbol-editing mode by doing one of the following:

- Select an instance of the symbol on the Stage and right-click (Windows) or Control-click (Macintosh), and select Edit.
- Select the symbol in the Library and right-click (Windows) or Control-click (Macintosh), and select Edit.
- Double-click the symbol in the Library.

2 To move the horizontal or vertical guides, drag and release a guide. The new position of the guide is updated in the Library preview for the symbol.

About runtime bitmap caching movie clip and button symbols

Runtime bitmap caching lets you optimize playback performance by specifying that a static movie clip (for example, a background image) or button symbol be cached as a bitmap at runtime. Caching a movie clip as a bitmap prevents Flash Player from having to continually redraw the image, which provides a significant improvement in playback performance.

For example, when you create animations with a complex background, create a movie clip for the background. The background is rendered as a bitmap stored at the current screen depth. It can be drawn quickly, letting the animation play faster and more smoothly.

Without bitmap caching, the animation might play back too slowly.

Bitmap caching lets you use a movie clip and freeze it in place automatically. If a region changes, vector data updates the bitmap cache. This process minimizes the number of redraws that Flash Player must perform, and provides smoother, faster playback performance.

Only use runtime bitmap caching on static, complex movie clips in which the position, but not the content, of the movie clip changes on each frame in an animation. The playback or runtime performance improvement from using runtime bitmap caching is only noticeable on complex-content movie clips. Runtime bitmap caching with simple movie clips does not enhance performance.

For more information, see When to enable caching in *Learning ActionScript 2.0 in Adobe Flash*.

***Note:** You can only use the Use Runtime Bitmap Caching option for movie clip and button symbols.*

Under the following circumstances, a movie clip does not use a bitmap (even if Use Runtime Bitmap Caching is selected) but instead renders the movie clip or button symbol by using vector data:

- The bitmap is too large (greater than 2880 pixels in either direction).
- The bitmap fails to allocate (producing an out of memory error).

Specify bitmap caching for a movie clip

1. Select the movie clip or button symbol on the Stage.
2. In the Property inspector, select Use Runtime Bitmap Caching.

Symbols and ActionScript

About controlling instances and symbols with ActionScript

To control movie clip and button instances, use ActionScript. The movie clip or button instance must have a unique instance name to be used with ActionScript. To control movie clip or button symbols, use ActionScript. For more information, see Handling events in *Learning ActionScript 2.0 in Adobe Flash* or Handling events in *Programming ActionScript 3.0*.

See also

“Edit symbols” on page 210

Controlling instances with behaviors

In FLA files where the ActionScript Publish setting is set to ActionScript 2.0, you can use behaviors to control movie clip and graphic instances in a document without writing ActionScript. Behaviors are prewritten ActionScript scripts that let you add ActionScript coding to your document without having to create the ActionScript code yourself. Behaviors are not available for ActionScript 3.0.

You can use behaviors with an instance to arrange it in the stacking order on a frame, as well as to load or unload, play, stop, duplicate, or drag a movie clip, or to link to a URL.

In addition, you can use behaviors to load an external graphic or an animated mask into a movie clip.

Flash includes the behaviors in the following table.

Behavior	Purpose	Select or input
Load Graphic	Loads an external JPEG file into a movie clip or screen.	Path and filename of JPEG file. Instance name of movie clip or screen receiving the graphic.
Load External Movieclip	Loads an external SWF file into a target movie clip or screen.	URL of external SWF file. Instance name of movie clip or screen receiving the SWF file.
Duplicate Movieclip	Duplicates a movie clip or screen.	Instance name of movie clip to duplicate. X-offset and Y-offset of pixels from original to copy.
Goto And Play at frame or label	Plays a movie clip from a particular frame.	Instance name of target clip to play. Frame number or label to play.
Goto And Stop at frame or label	Stops a movie clip, optionally moving the playhead to a particular frame.	Instance name of target clip to stop. Frame number or label to stop.
Bring To Front	Brings target movie clip or screen to the top of the stacking order.	Instance name of movie clip or screen.
Bring Forward	Brings target movie clip or screen one position higher in the stacking order.	Instance name of movie clip or screen.
Send To Back	Sends the target movie clip to the bottom of the stacking order.	Instance name of movie clip or screen.
Send Backward	Sends the target movie clip or screen one position lower in the stacking order.	Instance name of movie clip or screen.
Start Dragging Movieclip	Starts dragging a movie clip.	Instance name of movie clip or screen.
Stop Dragging Movieclip	Stops the current drag.	
Unload Movieclip	Removes a movie clip that was loaded by means of loadMovie() from Flash Player.	Instance name of movie clip.

See also

“Control sounds using behaviors” on page 298

“Control video playback using behaviors” on page 322

Add and configure a behavior

Be sure you are working in a FLA file whose ActionScript Publish setting is ActionScript 2.0 or earlier.

1 Select the object, such as a button, to trigger the behavior.

2 In the Behaviors panel (Window > Behaviors), click the Add (+) button and select the desired behavior from the Movieclip submenu.

3 Select the movie clip to control with the behavior.

4 Select a relative or absolute path.

5 If required, select or input settings for the behavior parameters and click OK. Default settings for the behavior appear in the Behaviors panel.

6 Under Event, click On Release (the default event) and select a mouse event from the menu. To use the On Release event, leave the option unchanged.

See also

"About relative paths" on page 72

"About absolute paths" on page 71

Create custom behaviors

To write custom behaviors, create an XML file that contains the ActionScript 2.0 code to perform the desired behavior, and save the file in the Behaviors folder of your local computer. Behaviors are stored in the following location:

- Windows: C:\Documents and Settings*user name*\Local Settings\Application Data\Adobe\Flash CS3*language*\Configuration\Behaviors
- Macintosh: Macintosh HD/Users/*user name*/Library/Application Support/Adobe/Flash CS3/*language*/Configuration/Behaviors/

Before you create your own behaviors, examine the Behavior XML files to develop an understanding of the syntax of the XML files, as well as the ActionScript code used to create behaviors. If you are new to writing behaviors, familiarize yourself with the XML tags used to create user interface elements (such as dialog boxes), and with ActionScript, the coding language used to create behaviors. To learn about the XML used to create interface elements, see *Extending Flash*. To learn about ActionScript, see *Programming ActionScript 3.0* or *Learning ActionScript 2.0 in Adobe Flash*.

You can also download behaviors that other Flash users have created from the Adobe Flash Exchange website. You can visit the Adobe Exchange at: www.adobe.com/go/flash_exchange.

1 Using an XML editor, open an existing behavior's XML file, and rename the file appropriately for the behavior you intend to create.

2 Enter a new value for the `category` attribute of the `behavior_devinition` tag in the XML file.

The following XML code creates a category named myCategory in the Flash Behaviors panel under which the behavior will be listed.

```
<behavior_definition dialogID="Trigger-dialog" category="myCategory"
authoringEdition="pro" name="behaviorName">
```

3 Enter a new value for the name attribute of the behavior_definition tag. This will be the name of the behavior as it will appear in the Flash authoring environment.

4 (Optional) If your custom behavior requires a dialog box, enter parameters using the `<properties>` and `<dialog>` tags.

To learn about the tags and parameters used to create your own custom dialog boxes, see *Extending Flash*.

5 In the `<actionscript>` tag, insert the ActionScript code to create the behavior.

If you are new to ActionScript, see *Learning ActionScript 2.0 in Adobe Flash* or *Programming ActionScript 3.0*.

For example (from the Movieclip_loadMovie.xml behavior file) (ActionScript 2.0):

```
<actionscript>
  <![CDATA[ //load Movie Behavior
    if($target$ == Number($target$)){
        loadMovieNum($clip$,$target$);
    } else {
        $target$.loadMovie($clip$);
    }
    //End Behavior
  ]]>
</actionscript>
```

6 Save the file and test the behavior.

See also

“Break apart an instance symbol” on page 215

Chapter 10: Creating animation

Adobe® Flash® CS3 Professional provides several ways to create animation and special effects; Timeline effects, tweened animation, changing the contents of successive frames in the Timeline, and frame-by-frame animation all provide you with different possibilities to create engaging, animated content.

Animation basics

Creating motion

Adobe® Flash® CS3 Professional offers several ways to include animation and special effects in a document, such as Timeline effects, tweened animation, changing the contents of successive frames in the Timeline, and frame-by-frame animation.

See also

"About tweened animation" on page 239

About layers in animation

Each scene in a Flash document can consist of any number of layers. Layers and layer folders organize the components of an animation sequence and separate animated objects so they don't erase, connect, or segment each other. To tween the movement of more than one group or symbol at once, each must be on a separate layer. Typically, the background layer contains static artwork, and each additional layer contains one separate animated object.

When a document has several layers, tracking and editing the objects on one or two of them can be difficult. This task is easier if you work with the contents of one layer at a time. Layer folders help you organize layers into manageable groups.

Creating keyframes

Changes in the animation are defined in a *keyframe*. When you create frame-by-frame animation, every frame is a keyframe. In tweened animation, you define keyframes at significant points in the animation and Flash creates the contents of frames between. The interpolated frames of a tweened animation appear as light blue or light green with an arrow drawn between keyframes. Because Flash documents save the shapes in each keyframe, create keyframes only at those points in the artwork where something changes.

Keyframes are indicated in the Timeline: a solid circle represents a keyframe with content on it, and an empty circle before the frame represents an empty keyframe. Subsequent frames added to the same layer have the same content as the keyframe.

Create keyframes

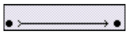
❖ Do one of the following:

- Select a frame in the Timeline, and select Insert > Timeline > Keyframe.
- Right-click (Windows) or Control-click (Macintosh) a frame in the Timeline and select Insert Keyframe.

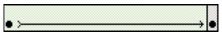
About representations of animation in the Timeline

Flash distinguishes tweened animation from frame-by-frame animation in the Timeline as follows:

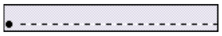
- A black dot at the beginning keyframe indicates motion tweens; a black arrow with a light blue background indicates intermediate tweened frames.



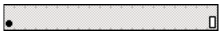
- A black dot at the beginning keyframe indicates shape tweens; a black arrow with a light green background indicates intermediate frames.



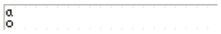
- A dashed line indicates that the tween is broken or incomplete, such as when the final keyframe is missing.



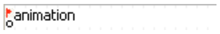
- A black dot indicates a single keyframe. Light gray frames after a single keyframe contain the same content with no changes and have a black line with a hollow rectangle at the last frame of the span.



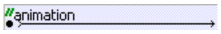
- A small *a* indicates that the frame is assigned a frame action with the Actions panel.



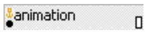
- A red flag indicates that the frame contains a label.



- A green double slash indicates that the frame contains a comment.



- A gold anchor indicates that the frame is a named anchor.



About frame rates

The frame rate, the speed the animation is played at, is measured in number of frames per second (fps). A frame rate that's too slow makes the animation appear to stop and start; a frame rate that's too fast blurs the details of the animation. A frame rate of 12 fps usually gives the best results on the web. QuickTime and AVI movies generally have a frame rate of 12 fps, while the standard motion-picture rate is 24 fps.

The complexity of the animation and the speed of the computer the animation is being played on affect the smoothness of the playback. To determine optimum frame rates, test your animations on a variety of machines.

Because you specify only one frame rate for the entire Flash document, set this rate before you begin creating animation.

See also

"Create or open a document and set its properties" on page 52

About frame-by-frame animation

Frame-by-frame animation changes the contents of the Stage in every frame and is best suited to complex animation in which an image changes in every frame instead of moving across the Stage. Frame-by-frame animation increases file size more rapidly than tweened animation. In frame-by-frame animation, Flash stores the values for each complete frame.

Create frame-by-frame animations

To create a frame-by-frame animation, define each frame as a keyframe and create a different image for each frame. Each new keyframe initially contains the same contents as the keyframe preceding it, so you can modify the frames in the animation incrementally.

1 Click a layer name to make it the active layer, and select a frame in the layer where the animation is to start.

2 If the frame isn't already a keyframe, select Insert > Timeline > Keyframe.

3 Create the artwork for the first frame of the sequence. Use the drawing tools, paste graphics from the Clipboard, or import a file.

4 To add a new keyframe whose contents are the same as those of the first keyframe, click the next frame to the right in the same row and select Insert > Timeline > Keyframe, or right-click (Windows) or Control-click (Macintosh) and select Insert Keyframe.

5 To develop the next increment of the animation, alter the contents of this frame on the Stage.

6 To complete your frame-by-frame animation sequence, repeat steps 4 and 5 until you've built the desired motion.

7 To test the animation sequence, select Control > Play or click the Play button on the Controller.

Edit animations

Only keyframes are editable. You can view tweened frames, but you can't edit them directly. To edit tweened frames, change one of the defining keyframes or insert a new keyframe between the beginning and ending keyframes. Drag items from the Library panel onto the Stage to add the items to the current keyframe.

To display and edit more than one frame at a time, use onion skinning.

Insert frames in the Timeline

- To insert a new frame, select Insert > Timeline > Frame.
- To create a new keyframe, select Insert > Timeline > Keyframe, or right-click (Windows) or Control-click (Macintosh) the frame where you want to place a keyframe, and select Insert Keyframe.
- To create a new blank keyframe, select Insert > Timeline > Blank Keyframe, or right-click (Windows) or Control-click (Macintosh) the frame where you want to place the keyframe, and select Insert Blank Keyframe.

Delete or modify a frame or keyframe

- To delete a frame, keyframe, or frame sequence, select it and right-click (Windows) or Control-click (Macintosh) and select Remove Frames. Surrounding frames remain unchanged.
- To move a keyframe or frame sequence and its contents, select it and drag to the desired location.

- To extend the duration of a keyframe, Alt-drag (Windows) or Option-drag (Macintosh) it to the final frame of the new sequence.
- To copy and paste a frame or frame sequence, select it and select Edit > Timeline > Copy Frames. Select a frame or sequence to replace, and select Edit > Timeline > Paste Frames.
- To convert a keyframe to a frame, select the keyframe and select Modify > Timeline > Clear Keyframe, or right-click (Windows) or Control-click (Macintosh) it and select Clear Keyframe. The cleared keyframe and all frames up to the subsequent keyframe are replaced with the contents of the frame preceding the cleared keyframe.
- To copy a keyframe or frame sequence by dragging, select it and Alt-drag (Windows) or Option-drag (Macintosh) to the new location.
- To change the length of a tweened sequence, drag the beginning or ending keyframe left or right.
- To add a library item to the current keyframe, drag the item from the Library panel onto the Stage.
- To reverse an animation sequence, select the appropriate frames in one or more layers and select Modify > Timeline > Reverse Frames. Keyframes must be at the beginning and end of the sequence.

Use onion skinning

Usually, one frame of the animation sequence at a time appears on the Stage. To help position and edit a frame-by-frame animation, view two or more frames on the Stage at once. The frame under the playhead appears in full color, while surrounding frames are dimmed, making it appear as if each frame were drawn on a sheet of translucent onion-skin paper and the sheets were stacked on top of each other. Dimmed frames cannot be edited.

Simultaneously view several frames of an animation on the Stage

❖ Click the Onion Skin button . All frames between the Start Onion Skin and End Onion Skin markers (in the Timeline header) are superimposed as one frame in the Document window.

Control onion skinning display

- To display onion skinned frames as outlines, click the Onion Skin Outlines button .
- To change the position of either onion skin marker, drag its pointer to a new location. (Normally, the onion skin markers move in conjunction with the current frame pointer.)
- To enable editing of all frames between onion skin markers, click the Edit Multiple Frames button . Usually, onion skinning lets you edit only the current frame. However, you can display the contents of each frame between the onion skin markers, and make each available for editing, regardless of which is the current frame.

***Note:** Locked layers (those with a padlock icon) aren't displayed when onion skinning is turned on. To avoid a multitude of confusing images, lock or hide the layers you don't want to be onion skinned.*

Change the display of onion skin markers

❖ Click the Modify Onion Markers button and select an item:

Always Show Markers Displays the onion skin markers in the Timeline header whether or not onion skinning is on.

Anchor Onion Locks the onion skin markers to their current position in the Timeline header. Usually, the onion skin range is relative to the current frame pointer and the onion skin markers. Anchoring the onion skin markers prevents them from moving with the current frame pointer.

Onion 2 Displays two frames on either side of the current frame.

Onion 5 Displays five frames on either side of the current frame.

Onion All Displays all frames on either side of the current frame.

Move an entire animation

Move the graphics in all frames and layers at once to avoid realigning everything.

1 Unlock all layers. To move everything on one or more layers but nothing on other layers, lock or hide all the layers you don't want to move.

2 Click the Edit Multiple Frames button ⧉.

3 Drag the onion skin markers so that they enclose all the frames to select, or click Modify Onion Markers and select Onion All.

4 Select Edit > Select All.

5 Drag the entire animation to the new location on the Stage.

Copy and paste a motion tween

Copy and paste motion lets you copy a motion tween, and paste (or apply) the frames, tween, and symbol information to another object.

1 Select the frames in the Timeline that contain the motion tween to copy. The frames you select must be on the same layer, however, they do not have to span a single motion tween. The selection can span a tween, empty frames, or two or more tweens.

2 Select Edit > Timeline > Copy Motion.

3 Select the symbol instance to receive the copied motion tween.

4 Select Edit > Timeline > Paste Motion.

The necessary frames, tween, and symbol information are inserted to match the original, copied tween.

To copy a symbol's motion tween to the Actions panel or use it in another project as ActionScript™, use the Copy Motion As ActionScript command.

Paste motion tween properties

Paste motion tween properties lets you copy a motion tween, and paste only specific properties to apply to another object.

1 Select the frames in the Timeline that contain the motion tween to copy. The frames you select must be on the same layer, however, they do not have to span a single motion tween. The selection can span a tween, empty frames, or two or more tweens.

2 Select Edit > Timeline > Copy Motion.

3 Select the symbol instance to receive the copied motion tween.

4 Select Edit > Timeline > Paste Motion Special. Select the specific motion tween properties to paste to the symbol instance. The tween properties are:

X Position How far an object moves in the x direction.

Y Position How far an object moves in the y direction.

Horizontal Scale The ratio between the object's current size and its natural size in the horizontal direction (X).

Vertical Scale Specifies the ratio between the object's current size and its natural size in the vertical direction (Y).

Rotation And Skew The rotation and skew of the object. These properties must be jointly applied to an object. Skew is a measurement of rotation in degrees, and when you rotate and skew, each property affects the other.

Color All color values such as Tint, Brightness, and Alpha are applied to the object.

Filters All filter values and changes for the selected span. If filters are applied to an object, the filter is pasted with all values intact, and its state (enabled or disabled) applies to the new object as well.

Blend Mode Applies the blend mode of the object.

Override Target Scale Properties When unchecked, specifies that all properties be pasted relative to the target object. When checked, this option overrides the scale properties of the target.

Override Target Rotation And Skew Properties When unchecked, specifies that all properties be pasted relative to the target object. When checked, the pasted properties override the existing rotation and scale properties of the object.

The necessary frames, tween, and symbol information are inserted to match the original, copied tween.

To copy a symbol's motion tween to the Actions panel or use it in another project as ActionScript, use the Copy Motion As ActionScript command.

Copy motion as ActionScript

Copy the properties that define a motion tween in the Timeline as ActionScript 3.0 and apply that motion to another symbol, either in the Actions panel or in the source files (such as class files) for a Flash document that uses ActionScript 3.0.

Use the fl.motion classes to customize the Flash-generated ActionScript for your specific project. For more information, see the fl.motion classes in the *ActionScript 3.0 Language and Components Reference*.

Copy Motion As ActionScript 3.0 can capture the following properties of a motion tween:

- Position
- Scale
- Skew
- Rotation
- Transformation Point
- Color
- Blend Mode
- Orientation To Path
- Scale
- Cache As Bitmap Setting
- Frame Labels
- Motion Guides
- Custom Easing
- Filters

For a video tutorial about copying and pasting motion as ActionScript, see www.adobe.com/go/vid0126.

1 Select the frames in the Timeline that contain the motion tween to copy. The frames you select must be on the same layer, however, they do not have to span a single motion tween. The selection can span a tween, empty frames, or two or more tweens.

2 Select Edit > Timeline > Copy Motion As ActionScript 3.0.

Provide the name of the instance to attach the motion tween to. Flash generates ActionScript specific to that instance name. Edit the Flash-generated ActionScript later for different instances.

Flash copies to your system's clipboard the ActionScript code that describes the selected motion tween. Paste the code into the Actions panel of a Flash document that contains the instance to receive the copied tween.

The ActionScript code contains the following items:

- An import of the fl.motion.Animator class
- An XML object for the provided instance (instance_xml:XML)
- An Animator object for the instance (instance_animator:Animator)
- A call to the Animator.play() method to begin the animation

***Note:** When working with motion guides, some differences might occur between the XML that the Copy Motion command and the Copy Motion As ActionScript 3.0 command generate. If a motion guide and custom easing is applied to a tween, the Copy Motion XML tags include the properties of the Bezier curves for the easing, and the XML uses only two keyframes. For the same tween, using the Copy Motion As ActionScript 3.0 command creates keyframes for each frame, and applies the correct values to each keyframe. If you remove the motion guide, the same XML code appears describing the custom easing for both commands.*

Extend still images

When you create a background for animation, a still image might remain the same for several frames. Adding a span of new frames (not keyframes) to a layer extends the contents of the last keyframe in all the new frames.

To manually extend a still image through multiple frames, or to use a shortcut, create an image in the first keyframe of the sequence and do one of the following:

- Select a frame to the right, marking the end of the span of frames to add and select Insert > Timeline > Frame.
- Alt-drag (Windows) or Option-drag (Macintosh) the keyframe to the right. This creates a span of new frames with a keyframe at the end point.

About mask layers

For spotlight effects and transitions, use a mask layer to create a hole through which underlying layers are visible. A mask item can be a filled shape, a type object, an instance of a graphic symbol, or a movie clip. Group multiple layers under a single mask layer to create sophisticated effects.

To create dynamic effects, animate a mask layer. For a filled shape used as a mask, use shape tweening; for a type object, graphic instance, or movie clip, use motion tweening. When using a movie clip instance as a mask, animate the mask along a motion path.

To create a mask layer, place a mask item on the layer to use as a mask. Instead of having a fill or stroke, the mask item acts as a window that reveals the area of linked layers beneath it. The rest of the mask layer conceals everything except what shows through the mask item. A mask layer can contain only one mask item. A mask layer cannot be inside a button, and you cannot apply a mask to another mask.

To create a mask layer from a movie clip, use ActionScript. A mask layer created with ActionScript can be applied only to another movie clip. See Using movie clips as masks in *Learning ActionScript 2.0 in Adobe Flash*.

For a video tutorial about animating masks, see www.adobe.com/go/vid0127.

For a sample of scriptable masks, see the Flash Samples web page at www.adobe.com/go/learn_fl_samples. Download and decompress the Samples zip file and navigate to the Masking\ScriptableMasks folder to access the sample.

Work with mask layers

You can use mask layers to reveal portions of a picture or graphic in the layer below. To create a mask, you specify that a layer is a mask layer, and either draw or place a filled shape on that layer. You can use any filled shape, including groups, text, and symbols, as a mask. The mask layer reveals the area of linked layers beneath the filled shape.

For a video tutorial about animating masks, see www.adobe.com/go/vid0127.

For a sample of scriptable masks, see the Flash Samples web page at www.adobe.com/go/learn_fl_samples. Download and decompress the Samples zip file and navigate to the Masking\ScriptableMasks folder to access the sample.

See also

"Tweened animation" on page 239

"Add a motion tween to instances, groups, or type" on page 241

"Tween motion along a path" on page 244

Create a mask layer

1 Select or create a layer containing the objects to appear inside the mask.

2 Select Insert > Timeline > Layer to create a new layer above it. A mask layer always masks the layer immediately below it; create the mask layer in the proper place.

3 Place a filled shape, text, or an instance of a symbol on the mask layer. Flash ignores bitmaps, gradients, transparency, colors, and line styles in a mask layer. Any filled area is completely transparent in the mask; any nonfilled area is opaque.

4 Right-click (Windows) or Control-click (Macintosh) the mask layer's name in the Timeline, and select Mask. A mask layer icon indicates the mask layer. The layer immediately below it is linked to the mask layer, and its contents show through the filled area on the mask. The masked layer name is indented, and its icon changes to a masked layer icon.

5 To display the mask effect in Flash, lock the mask layer and the masked layer.

Mask additional layers after creating a mask layer

- Drag an existing layer directly below the mask layer.
- Create a new layer anywhere below the mask layer.
- Select Modify > Timeline > Layer Properties, and select Masked.

Unlink layers from a mask layer

❖ Select the layer to unlink and do one of the following:

- Drag the layer above the mask layer.

- Select Modify > Timeline > Layer Properties, and select Normal.

Animate a filled shape, type object, or graphic symbol instance on a mask layer

1 Select the mask layer in the Timeline.

2 To unlock the mask layer, click in the Lock column.

3 Do one of the following:

- If the mask object is a filled shape, apply shape tweening to the object.
- If the mask object is a type object or graphic symbol instance, apply motion tweening to the object.

4 When the animation operation is complete, click in the Lock column for the mask layer to relock the layer.

Animate a movie clip on a mask layer

1 Select the mask layer in the Timeline.

2 To edit the movie clip in place and to display the movie clip's Timeline, double-click the movie clip on the Stage.

3 Apply motion tweening to the movie clip.

4 When the animation procedure is complete, click the Back button to return to document-editing mode.

5 To lock the layer again, click in the Lock column for the mask layer.

Using Timeline effects

About Timeline effects

Prebuilt Timeline effects let you create complex animations with a minimal number of steps. You can apply Timeline effects to the following objects:

- Text
- Graphics, including shapes, groups, and graphic symbols
- Bitmap images
- Button symbols

***Note:** When you apply a Timeline effect to a movie clip, the effect is nested in the movie clip.*

Add a Timeline effect

When you add a Timeline effect to an object, Flash creates a layer and transfers the object to the new layer. The object is placed inside the effect graphic, and all tweens and transformations required for the effect reside in the graphic on the newly created layer.

The new layer automatically receives the same name as the effect, appended with a number that represents the order in which the effect is applied, out of all effects in your document.

When you add a Timeline effect, a folder with the effect's name is added to the library, containing elements used in creating the effect.

1 Do one of the following:

- Select the object to add the Timeline effect to. Select Insert > Timeline Effects, make a selection, and select an effect from the list.
- Right-click (Windows) or Control-click (Macintosh) the object to add the Timeline effect to. Select Timeline Effects, make a selection, and select an effect.

2 View the effect preview based on default settings. Modify the default settings, and click Update Preview to view the effect with the new settings.

3 When the desired Timeline effect appears in the preview window, click OK.

Timeline effect settings

Each Timeline effect manipulates a graphic or symbol in a specific way and allows you to change individual parameters for a desired effect. The preview window shows the changes made when you alter settings.

Motion effect name and description	**Settings**
Copy to grid	
Duplicates a selected object by the number of columns and then multiplies the columns by the number of rows to create a grid of the elements.	• Number of rows • Number of columns • Distance between rows, in pixels • Distance between columns, in pixels
Distributed duplication	
Duplicates a selected object the number of times entered in the settings. The first element is a copy of the original object. The objects are modified in increments until the final object reflects the parameters entered in the settings.	• Number of copies • Offset distance, x position, in pixels • Offset distance, y position, in pixels • Offset rotation, in degrees • Offset start frame, in frames across Timeline • Exponential scaling by x, y scale, in delta percentage • Linear scaling by x, y scale, in delta percentage • Final alpha, in percentage • Change color, select/deselect • Final color, in RGB hex value (final copy has this color value; intermediate copies gradually transition to it) • Duplication delay, in frames (results in pause between copies)
Blur	

Motion effect name and description	**Settings**
Creates a motion blur effect by changing the alpha value, position, or scale of an object over time.	• Effect duration, in frames • Allow horizontal blur • Allow vertical blur • Direction of blur • Number of steps • Starting scale
Drop shadow	
Creates a shadow below the selected element.	• Color, in hex RGB value • Alpha transparency, in percentage • Shadow offset, in *x, y* offset, in pixels
Expand	
Expands, contracts, or expands and contracts objects over time. This effect works best with two or more objects grouped together or combined in a movie clip or graphic symbol. Objects containing text or letters work well with this effect.	• Expand duration, in frames • Expand, squeeze, both • Expand direction, to left, from center, to right • Fragment offset, in pixels • Shift group center by, *x, y* offset, in pixels • Change fragment size by, height, width, in pixels
Explode	
Gives the illusion of an object exploding. Elements of text or a complex group of objects (symbols, shapes or video clips) break apart, spin, and arc outward.	• Effect duration, in frames • Direction of explosion, upward to left, center, or right, downward to left, center, or right • Arc size, *x, y* offset in pixels • Rotate fragments by, in degrees • Change fragments size by, in degrees • Final alpha, in percentage
Transform	

Motion effect name and description	Settings
Adjusts the position, scale, rotation, alpha, and tint of the selected elements. Use Transform to apply a single effect or a combination of effects to create Fade In/Out, Fly In/Out, Grow/Shrink, and Spin Left/Right effects.	• Effect duration, in frames • Move to position, *x, y* offset, in pixels • Change position by, *x, y* offset, in pixels • Scale, lock to equally apply change, in percentage, unlock to apply *x* and/or *y* axis change separately, in percentage • Rotate, in degrees • Spin, number of times • Times, counterclockwise, clockwise • Change color, select/deselect • Final color, in RGB hex value • Final alpha, in percentage • Motion ease
Transition	
Wipes in or wipes out selected objects by fading, wiping, or a combination of both.	• Effect duration, in frames • Direction, toggle between in (coming in) and out (going out), select up, down, left, or right • Fade, select/deselect • Wipe, select/deselect • Motion ease

Edit a Timeline effect

1 Select the object associated with the effect on the Stage and do one of the following:

- In the Property inspector, click Edit.
- Right-click (Windows) or Control-click (Macintosh) the object, and select Timeline Effects > Edit Effect.

2 Edit the settings and click OK.

Delete a Timeline effect

❖ On the Stage, right-click (Windows) or Control-click (Macintosh) the object that has the Timeline effect to remove, and select Timeline Effects > Remove Effect.

Tweened animation

About tweened animation

Flash can create two types of tweened animation:

- In motion tweening, you define properties such as position, size, and rotation for an instance, group, or text block at one specific time, and change those properties at another specific time. You can also apply a motion tween along a path.

- In shape tweening, you draw a shape at one specific time, and change that shape or draw another shape at another specific time. Flash interpolates the values or shapes for the frames in between, creating the animation.

***Note:** To apply shape tweening to groups, instances, or bitmap images, break these elements apart. To apply shape tweening to text, break the text apart twice to convert the text to objects.*

Tweened animation is an effective way to create movement and changes over time while minimizing file size. In tweened animation, only the values for the changes between frames are stored.

To quickly prepare elements in a document for tweened animation, distribute objects to layers.

To create a dynamic mask, apply tweened animation to an object on a mask layer.

For video tutorials about animation, see the following:

- Creating shape tweens: www.adobe.com/go/vid0124
- Creating motion tweens: www.adobe.com/go/vid0125
- Copying and pasting ActionScript from an animation: www.adobe.com/go/vid0126
- Creating and animating masks: www.adobe.com/go/vid0127

For a text tutorial about timeline animation, see Create a Timeline Animation on the Flash Tutorials page at www.adobe.com/go/learn_fl_tutorials.

For samples of animation, see the Flash Samples page at www.adobe.com/go/learn_fl_samples. The following samples are available:

- Animated Drop Shadow: Download and decompress the Samples zip file and navigate to the Graphics\AnimatedDropShadow folder to access the sample.
- Animation and Gradients: Download and decompress the Samples zip file and navigate to the Graphics\AnimationAndGradients folder to access the sample.

See also

"About mask layers" on page 234

"Break text apart" on page 270

"Break apart groups and objects" on page 202

Distributing objects to layers for tweened animation

To separate layers to apply tweened animation to the objects, distribute selected objects in a frame. Flash distributes each object to a new, separate layer. Any objects that you don't select (including objects in other frames) are preserved in their original positions.

You can apply the Distribute To Layers command to any type of element on the Stage, including graphic objects, instances, bitmaps, video clips, and broken-apart text blocks.

See also

"Break text apart" on page 270

About new layers

New layers created during the Distribute To Layers operation are named according to the name of the element that each contains:

- A new layer containing a library asset (such as a symbol, bitmap, or video clip) receives the same name as the asset.
- A new layer containing a named instance receives the name of the instance.
- A new layer containing a character from a broken-apart text block is named with the character.
- A new layer containing a graphic object (which has no name) is named Layer1 (or Layer2, and so on), because graphic objects do not have names.

Flash inserts new layers below any selected layers. The new layers are arranged top to bottom, in the order that the selected elements were originally created. The layers in broken-apart text are arranged in the order of the characters, whether left-to-right, right-to-left, or top-to-bottom. For example, if you break apart the text *FLASH* and distribute it to layers, the new layers, named F, L, A, S, and H, are arranged top to bottom, immediately below the layer that initially contained the text.

Distribute objects to layers

1 Select the objects to distribute to layers. The objects can be in a single layer, or in several layers, including noncontiguous layers.

2 Do one of the following:

- Select Modify > Timeline > Distribute To Layers.
- Right-click (Windows) or Control-click (Macintosh) one of the selected objects, and select Distribute To Layers.

See also

“Tweened animation” on page 239

Add a motion tween to instances, groups, or type

To tween the changes in properties of instances, groups, and type, use motion tweening. Flash can tween position, size, rotation, and skew of instances, groups, and type. Additionally, Flash can tween the color of instances and type, creating gradual color shifts or making an instance fade in or out. To tween the color of groups or type, make them into symbols. To animate individual characters in a block of text separately, place each character in a separate text block.

If you apply a motion tween and then change the number of frames between the two keyframes, or move the group or symbol in either keyframe, Flash automatically tweens the frames again.

For a video tutorial about motion tweens, see www.adobe.com/go/vid0125.

For a text tutorial about motion tweens, see Create a Timeline Animation on the Flash Tutorials page at www.adobe.com/go/learn_fl_tutorials.

For samples of animation, see the Flash Samples page at www.adobe.com/go/learn_fl_samples. The following samples are available:

- Animated Drop Shadow: Download and decompress the Samples zip file and navigate to the Graphics\AnimatedDropShadow folder to access the sample.
- Animation and Gradients: Download and decompress the Samples zip file and navigate to the Graphics\AnimationAndGradients folder to access the sample.

See also

“Create symbols” on page 208

“Break text apart” on page 270

“Tweened animation” on page 239

Create a motion tween by using the Motion Tweening option

1 Click a layer name to make it the active layer, and select an empty keyframe in the layer where you want the animation to start.

2 To create the first frame of the motion tween, do one of the following:

- Create a graphic object with the Pen, Oval, Rectangle, Pencil, or Brush tool, and then convert it to a symbol.
- Create an instance, group, or text block on the Stage.
- Drag an instance of a symbol from the Library panel.

3 Create a second keyframe where you want the animation to end, and leave the new keyframe selected.

4 To modify the item in the ending frame, do any of the following. To tween the color of elements other than instances or text blocks, use shape tweening:

- Move the item to a new position.
- Modify the item’s size, rotation, or skew.
- Modify the item’s color (instance or text block only).

5 Click any frame in the tween’s frame span and select Motion from the Tween pop-up menu in the Property inspector (Window > Properties > Properties).

6 If you modified the size of the item in step 4, select Scale to tween the size of the selected item.

7 To produce a more realistic sense of motion, apply easing to the motion tween. To apply easing to a motion tween, use the Ease slider to specify an easing value for each motion tween you create, or use the Custom Ease In/Ease Out dialog box to more precisely control the speed of the motion tween.

Drag the arrow next to the Easing value or enter a value to adjust the rate of change between tweened frames:

- To begin the motion tween slowly and accelerate the tween toward the end of the animation, drag the slider up or enter a negative value between -1 and -100.
- To begin the motion tween rapidly and decelerate the tween toward the end of the animation, drag the slider down or enter a positive value between 1 and 100.
- To produce a more complex change in speed within the tween’s frame span, use the Custom Ease In/Ease Out dialog box.

By default, the rate of change between tweened frames is constant. Easing creates a more natural appearance of acceleration or deceleration by gradually adjusting the rate of change.

8 To rotate the selected item while tweening, select an option from the Rotate menu:

- To prevent rotation, select None (the default setting).
- To rotate the object once in the direction requiring the least motion, select Auto.
- To rotate the object as indicated, and then enter a number to specify the number of rotations, select Clockwise (CW) or Counterclockwise (CCW).

Note: *The rotation in step 8 is in addition to any rotation you applied to the ending frame in step 4.*

9 If you’re using a motion path, select Orient To Path to orient the baseline of the tweened element to the motion path.

10 To synchronize the animation of graphic symbol instances with the main Timeline, select the Sync option in the Property inspector.

***Note:** Modify > Timeline > Synchronize Symbols and the Sync option both recalculate the number of frames in a tween to match the number of frames allotted to it in the Timeline.*

11 If you’re using a motion path, select Snap to attach the tweened element to the motion path by its registration point.

Create a motion tween by using the Create Motion Tween command

1 Select an empty keyframe and draw an object on the Stage, or drag an instance of a symbol from the Library panel.

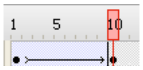
***Note:** To create a tween, you must have only one item on the layer.*

2 Select Insert > Timeline > Create Motion Tween. If you drew an object in step 1, Flash automatically converts the object to a symbol and names it *tween1*.

3 Click inside the frame where you want the animation to end, and select Insert > Timeline > Frame.

4 Move the item on the Stage to the desired position. Adjust the size of the element to tween its scale. To tween its rotation, adjust the rotation of the element. Deselect the object when you complete adjustments.

A keyframe is automatically added to the end of the frame range.



5 To adjust the rate of change between tweened frames, drag the arrow next to the Easing value or enter a value:

- To begin the motion tween slowly and accelerate the tween toward the end of the animation, drag the slider up or enter a value between -1 and -100.
- To begin the motion tween rapidly and decelerate the tween toward the end of the animation, drag the slider down or enter a positive value between 1 and 100. By default, the rate of change between tweened frames is constant. Easing creates a more natural appearance of acceleration or deceleration by gradually adjusting the rate of change.
- To produce a more complex change in speed within the tween’s frame span, use the Custom Ease In/Ease Out dialog box.

6 To rotate the selected item while tweening, select an option from the Rotate menu:

- To rotate the object once in the direction requiring the least motion, select Auto.
- To rotate the object as indicated, and then enter a number to specify the number of rotations, select Clockwise (CW) or Counterclockwise (CCW).

***Note:** The rotation in step 6 is in addition to any rotation you applied to the ending frame in step 4.*

7 If you’re using a motion path, select Orient To Path to orient the baseline of the tweened element to the motion path.

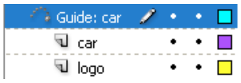
8 Select Synchronize to ensure that the instance loops properly in the main document.

Use the Synchronize command if the number of frames in the animation sequence inside the symbol is not an even multiple of the number of frames the graphic instance occupies in the document.

9 If you’re using a motion path, select Snap to attach the tweened element to the motion path by its registration point.

Tween motion along a path

Motion guide layers let you draw paths along which tweened instances, groups, or text blocks can be animated. You can link multiple layers to a motion guide layer to have multiple objects follow the same path. A normal layer that is linked to a motion guide layer becomes a guided layer.



In this example, two objects on separate layers are attached to the same motion path.

For a video tutorial about motion tweens, see www.adobe.com/go/vid0125.

For a tutorial about motion tweens, see Create a Timeline Animation on the Flash Tutorials web page at www.adobe.com/go/learn_fl_tutorials.

For samples of animation, see the Flash Samples page at www.adobe.com/go/learn_fl_samples. The following samples are available:

- Animated Drop Shadow: Download and decompress the Samples zip file and navigate to the Graphics\AnimatedDropShadow folder to access the sample.
- Animation and Gradients: Download and decompress the Samples zip file and navigate to the Graphics\AnimationAndGradients folder to access the sample.

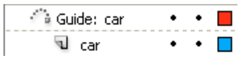
Create a motion path for a tweened animation

1 Create a motion-tweened animation sequence. If you select Orient To Path, the baseline of the tweened element orients to the motion path. If you select Snap, the registration point of the tweened element snaps to the motion path.

2 Do one of the following:

- Select the layer containing the animation, and select Insert > Timeline > Motion Guide.
- Right-click (Windows) or Control-click (Macintosh) the layer containing the animation and select Add Motion Guide.

Flash creates a new layer above the selected layer with a motion guide icon to the left of the layer name.



Motion guide layer above selected layer

3 Use the Pen, Pencil, Line, Circle, Rectangle, or Brush tool to draw the desired path.

4 Snap the center to the beginning of the line in the first frame, and to the end of the line in the last frame.

***Note:** For best snapping results, drag the symbol by its registration point.*

5 To hide the motion guide layer and the line so that only the object's movement is visible while you work, click in the Eye column on the motion guide layer.

The group or symbol follows the motion path when you play the animation.

Link layers to a motion guide layer

❖ Do one of the following:

- Drag an existing layer below the motion guide layer. The layer is indented under the motion guide layer. All objects on this layer automatically snap to the motion path.
- Create a new layer under the motion guide layer. Objects you tween on this layer are automatically tweened along the motion path.
- Select a layer below a motion guide layer. Select Modify > Timeline > Layer Properties, and select Guide.

Unlink layers from a motion guide layer

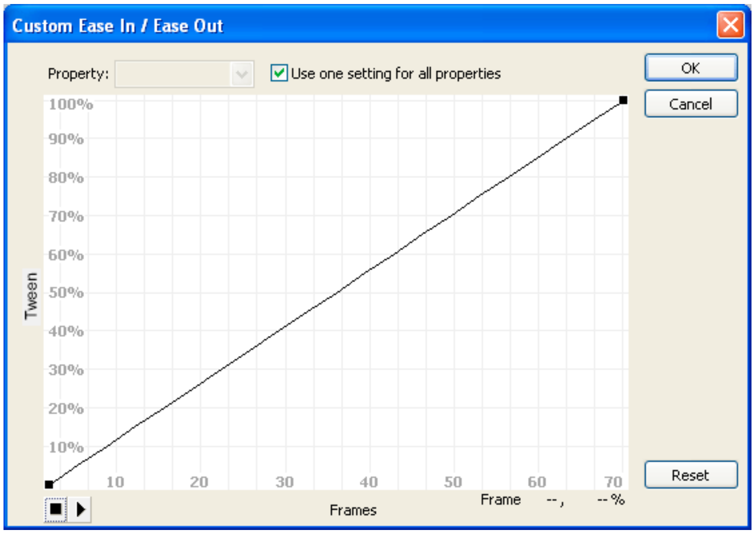
❖ Select the layer to unlink and do one of the following:

- Drag the layer above the motion guide layer.
- Select Modify > Timeline > Layer Properties, and select Normal as the layer type.

Apply custom ease in/ease out to motion tweens

The Custom Ease In/Ease Out dialog box displays a graph representing the degree of motion over time. The horizontal axis represents frames, and the vertical axis represents percentage of change. The first keyframe is represented as 0%, and the last keyframe is represented as 100%.

The slope of the graph's curve represents the rate of change of the object. When the curve is horizontal (no slope), the velocity is zero; when the curve is vertical, an instantaneous rate of change occurs.



Custom Ease In / Ease Out

Property:

☑ Use one setting for all properties

Custom Ease In/Ease Out graph showing constant velocity

Additional controls for the Custom Ease In/Ease Out dialog box

Use One Setting For All Properties check box The default is selected; the displayed curve is used for all properties, and the Properties pop-up menu is disabled. When the check box is not selected, the Properties pop-up menu is enabled, and each property has a separate curve defining the velocity of that property.

Property pop-up menu Enabled only when the Use One Setting for All Properties check box is not selected. When enabled, a separate curve is maintained for each of the five properties appearing in the menu. Selecting a property in the menu displays the curve for that property. The properties are:

- **Position** Specifies custom ease settings for the position of an animated object on the Stage.

- **Rotation** Specifies custom ease settings for the rotation of an animated object. For example, you can fine-tune how quickly or slowly an animated character turns around to face the user on the Stage.

- **Scale** Specifies custom ease settings for the scale of an animated object. For example, you can more easily customize the scale of an object so it appears to be moving away from the viewer, then coming closer, and then moving away again.

- **Color** Specifies custom ease settings for color transitions applied to an animated object.

- **Filters** Specifies custom ease settings for filters applied to an animated object. For example, you can control the ease setting of a drop shadow that simulates a change in the direction of a light source.

Play and Stop buttons Let you preview an animation on the Stage using all the current velocity curves defined in the Custom Ease In/Ease Out dialog box.

Reset button Lets you reset the velocity curve to the default, linear state.

Position of the selected control point In the lower-right corner of the dialog box, a numeric value displays the keyframe and position of the selected control point. If no control point is selected, no value appears.

To add a control point to the line, click the diagonal line once. To achieve a precise degree of control over the motion of an object, drag the positions of the control points.

Using frame indicators (represented by square handles), click where you want an object to slow down or speed up. Clicking the square handle of a control point selects that control point, and displays the tangent points on either side of it. Hollow circles represent tangent points. Drag the control point or its tangent points with the mouse or use the keyboard’s arrow keys to position them.

By default, the control points snap to a grid. You can turn off snapping by pressing the X key while dragging the control point.

Clicking an area of the curve away from any control points adds a new control point to the curve at that point, without changing the shape of the curve. Clicking away from the curve and control points deselects the currently selected control point.

Use the Custom Ease In/Ease Out dialog box

1 Select a layer in the Timeline that has a motion tween applied to it.

2 Click the Edit button next to the Ease slider in the frame Property inspector.

3 (Optional) To display the curve for a property, deselect Use One Setting For All Properties, and select a property in the menu.

4 To add a control point, Control-click (Windows) or Command-click (Macintosh) the diagonal line.

5 To increase the speed of the object, drag the control point up; to slow down the speed of the object, drag it downwards.

6 To further adjust the ease curve, and fine tune the ease value of the tween, drag the vertex handles.

7 To view the animation on the Stage, click the play button in the lower-left corner.

8 Adjust the controls until you achieve the desired effect.

Copy and paste an ease curve

- To copy the current ease curve, press Control+C (Windows) or Command+C (Macintosh).
- To paste the copied curve into another ease curve, press Control+V (Windows) or Command+V (Macintosh).

You can copy and paste the ease curve. The copied curve remains available until you exit the Flash application.

Compatibility with current ease in/ease out settings

If you use the Custom Ease In/Ease Out dialog box to apply a custom ease to a frame, the edit box that shows the ease value shows '--'. If you use the Edit box or the pop-up slider to apply an ease value to a frame, the Custom Ease graph is set to the equivalent curve, and the Use One Setting For All Properties check box is selected.

Unsupported easing curves

Certain types of easing curves are not supported. No part of the graph can represent a nonlinear curve (such as a circle).

The Custom Ease dialog box automatically prevents moving a control point or a tangent handle to a position that would render an invalid curve:

- All points must exist on the graph. Control points cannot be moved beyond the bounds of the graph.
- All segments of the curve must exist within the graph. The shape of the curve is flattened to prevent it from extending beyond the bounds of the graph.

Control shape changes with shape hints

To control more complex or improbable shape changes, you can use shape hints. Shape hints identify points that should correspond in starting and ending shapes. For example, if you are tweening a drawing of a face as it changes expression, you can use a shape hint to mark each eye. Then, instead of the face becoming an amorphous tangle while the shape change takes place, each eye remains recognizable and changes separately during the shift.

Shape hints contain letters (a through z) for identifying which points correspond in the starting and ending shape. You can use up to 26 shape hints.

Shape hints are yellow in a starting keyframe, green in an ending keyframe, and red when not on a curve.

For best results when tweening shapes, follow these guidelines:

- In complex shape tweening, create intermediate shapes and tween them instead of just defining a starting and ending shape.
- Make sure that shape hints are logical. For example, if you're using three shape hints for a triangle, they must be in the same order on the original triangle and on the triangle to be tweened. The order cannot be *abc* in the first keyframe and *acb* in the second.
- Shape hints work best if you place them in counterclockwise order beginning at the top-left corner of the shape.

For a video tutorial about shape tweens, see www.adobe.com/go/vid0124.

Use shape hints

1 Select the first keyframe in a shape-tweened sequence.

2 Select Modify > Shape > Add Shape Hint. The beginning shape hint appears as a red circle with the letter *a* somewhere on the shape.

3 Move the shape hint to a point to mark.

4 Select the last keyframe in the tweening sequence. The ending shape hint appears somewhere on the shape as a green circle with the letter *a*.

5 Move the shape hint to the point in the ending shape that should correspond to the first point you marked.

6 To view how the shape hints change the shape tweening, play the animation again. To fine-tune the tweening, move the shape hints.

7 Repeat this process to add additional shape hints. New hints appear with the letters that follow (*b*, *c*, and so on).

View all shape hints

❖ Select View > Show Shape Hints. The layer and keyframe that contain shape hints must be active for Show Shape Hints to be available.

Remove a shape hint

❖ Drag it off the Stage.

Remove all shape hints

❖ Select Modify > Shape > Remove All Hints.

Chapter 11: Special effects

Adobe® Flash® CS3 Professional special effects include filters, which let you add visual effects to text, buttons, and movie clips; and blend modes, which let you create composite images.

About filters

Filter overview

Adobe® Flash® CS3 Professional filters (graphic effects) let you add interesting visual effects to text, buttons, and movie clips. A feature unique to Flash is that you can animate the filters you apply using motion tweens.

Flash blend modes let you create composite images. *Compositing* is the process of varying the transparency or color interaction of two or more overlapping objects. Blending modes also add a dimension of control to the opacity of objects and images. You can use Flash blending modes to create highlights or shadows that let details from an underlying image show through, or to colorize a desaturated image.

About animating filters

You animate filters in the Timeline. Objects on separate keyframes joined by a tween have the parameters for corresponding filters tweened on intermediate frames. If a filter does not have a matching filter (a filter of the same type) at the opposite end of the tween, a matching filter is added automatically to ensure that the effect occurs at the end of the animation sequence.

To prevent motion tweens from functioning incorrectly if a filter is missing at one end of the tween, or if filters are applied in a different order at each end, Flash does the following:

- If you apply a motion tween to a movie clip with filters applied to it, when you insert a keyframe at the opposite end of the tween, the movie clip automatically has the same filters, with the same stacking order, on the last frame of the tween as it did at the beginning of the tween.
- If you place movie clips on two different frames with different filters applied to each, and you apply a motion tween between the frames, Flash first processes the movie clip with the most filters. Flash then compares the filters applied to the first movie clip against the filters that the second movie clip uses. If no matching filters are found in the second movie clip, Flash generates a dummy filter with no parameters and the color of the existing filters.
- If a motion tween exists between two keyframes and you add a filter to the object in one keyframe, Flash automatically adds a dummy filter to the movie clip when it reaches the keyframe at the other end of the tween.
- If a motion tween exists between two keyframes and you remove a filter from an object in one keyframe, Flash automatically removes the matching filter from the movie clip when it reaches the keyframe at the other end of the tween.
- If you set filter parameters inconsistently between the beginning and end of a motion tween, Flash applies the filter settings of the starting frame to the interpolated frames. Inconsistent settings occur when the following parameters are set differently between the beginning and end of the tween: knockout, inner shadow, inner glow, and type of gradient glow and gradient bevel.

For example, if you create a motion tween using the drop shadow filter, and apply a drop shadow with a knockout on the first frame of the tween, and an inner shadow on the last frame of the tween, Flash corrects the inconsistent use of the filter in the motion tween. In this case, Flash applies the filter settings used on the first frame of the tween—a drop shadow with a knockout.

About filters and Flash Player performance

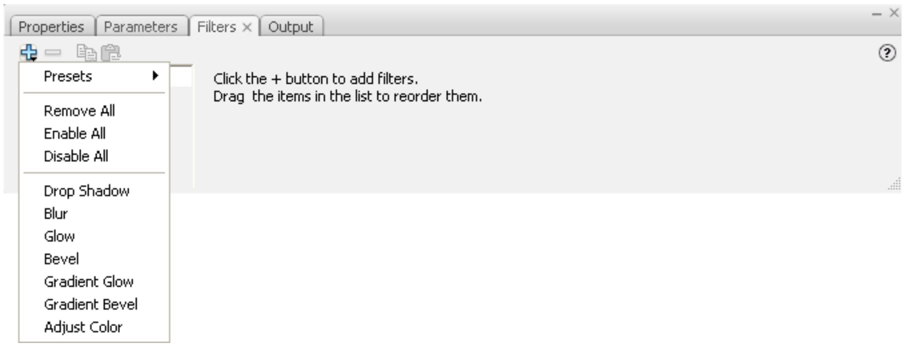
The type, number, and quality of filters you apply to objects can affect the performance of SWF files as you play them. The more filters you apply to an object, the greater the number of calculations Adobe® Flash® Player must process to correctly display the visual effects you've created. Adobe® recommends that you apply a limited number of filters to a given object.

Each filter includes controls that let you adjust the strength and quality of the applied filter. Using lower settings improves performance on slower computers. If you are creating content for playback on a wide range of computers, or are unsure of the computing power available to your audience, set the quality level to Low to maximize playback performance.

Apply filters

Each time you add a new filter to an object, it is added to the list of applied filters for that object in the Property inspector. You can apply multiple filters to an object, as well as remove filters that were previously applied. You can apply filters only to text, button, and movie clip objects.

You can create a filter settings library that lets you easily apply the same filter or sets of filters to an object. Flash stores the filter presets you create in the Property inspector on the Filters tab in the Filters > Presets menu.



Click the + button to add filters.
Drag the items in the list to reorder them.

The Add Filter menu in the Property inspector.

Apply or remove a filter

1 Select a text, button, or movie clip object to apply a filter to or remove a filter from.

2 Select Filters, and do one of the following:

- To add a filter, click the Add Filter (+) button, and select a filter. Experiment with the settings until you get the desired look.
- To remove a filter, select the filter to remove in the list of applied filters, and click the Remove Filter (-) button. You can delete or rename any presets.

Copy and paste a filter

1 Select the object to copy a filter from, and select the Filters panel.

2 Select the filter to copy, and click the Copy Filter button.

3 Select the object to apply the filter to, and click the Paste Filter button 📋.

Apply a preset filter to an object

1 Select the object to apply a filter preset to, and select the Filter tab.

2 Click the Add Filter (+) button, and select Presets.

3 Select the filter preset to apply from the list of available presets at the bottom of the preset menu.

Note: *When you apply a filter preset to an object, Flash replaces any filters currently applied to the selected objects with the filters used in the preset.*

Enable or disable a filter applied to an object

❖ Click the enable or disable icon next to the filter name in the Filter list.

Note: *To toggle the enable state of the other filters in the list, Alt-click (Windows) or Option-click (Macintosh) the enable icon in the Filter list. If you Alt-click the disable icon, the selected filter is enabled, and all others filters in the list are disabled.*

Enable or disable all filters applied to an object

❖ Click the Add Filter (+) button, and select Enable All or Disable All.

Note: *To enable or disable all of the filters in the list, Control-click the enable or disable icon in the Filter list.*

Create preset filter libraries

Save filter settings as preset libraries that you can easily apply to movie clip and text objects. Share your filter presets with other users by providing them with the filter configuration file. The filter configuration file is an XML file that is saved in the Flash Configuration folder in the following location:

- Windows: C:\Documents and Settings*username*\Local Settings\Application Data\Adobe\Flash CS3*language*\Configuration\Effects\filtername.xml
- Macintosh: Macintosh HD/Users/*username*/Library/Application Support/Adobe/Flash CS3/*language*/Configuration/Effects/*filtername.xml*

Create a library of filters with preset settings

1 Apply the filter or filters to the object.

2 Click the Add Filter (+) button, and select Presets > Save As.

3 Enter a name for the filter settings in the Save Preset As dialog box, and click OK.

Rename a filter preset

1 Click the Add Filter (+) button, and select Presets > Rename.

2 Double-click the preset name to modify.

3 Enter a new preset name, and click Rename.

Delete a filter preset

1 Click the Add Filter (+) button, and select Presets > Delete.

2 Select the preset to remove, and click Delete.

Apply a drop shadow

The Drop Shadow filter simulates the look of an object casting a shadow onto a surface.

Text with the Drop Shadow filter applied.

For a sample of a drop shadow, see the Flash Samples page at www.adobe.com/go/learn_fl_samples. Download and decompress the Samples zip file and navigate to the Graphics\AnimatedDropShadow directory.

1 Select the object to apply a drop shadow to, and select Filters.

2 Click the Add Filter (+) button, and select Drop Shadow.

3 Edit the filter settings on the Filter tab:

- To set the width and height of the drop shadow, drag the Blur X and Y sliders.
- To set the distance of the shadow from the object, drag the Distance slider.
- To open the Color Picker and set the shadow color, click the color control.
- To set the darkness of the shadow, drag the Strength slider. The higher the numerical value, the darker the shadow.
- To set the angle of the shadow, enter a value, or click the angle selector and drag the angle dial.
- Select Knockout to knock out (or visually hide) the source object and display only the drop shadow on the knockout image.
- To apply the shadow within the boundaries of the object, select Inner shadow.
- To hide the object and display only its shadow, select Hide Object. Hide Object lets you more easily create a realistic shadow.
- Select the quality level for the drop shadow. High is approximate to that of a Gaussian blur. Low maximizes playback performance.

Create a skewed drop shadow

Skewing the Drop Shadow filter to create a more realistic looking shadow.

1 Select the object with the shadow you want to skew.

2 Duplicate (select Edit > Duplicate) the source object.

3 Select the duplicated object, and skew it using the Free Transform tool (Modify > Transform > Rotate And Skew).

4 Apply the Drop Shadow filter to the duplicated movie clip or text object. (It will already be applied if the object you duplicated already had a drop shadow.)

5 In the Filters panel, select Hide Object to hide the duplicated object while leaving its shadow visible.

6 Select Modify > Arrange > Send Backward to place the duplicated object and its shadow behind the original object that you duplicated.

7 Adjust both the Drop Shadow filter settings and the angle of the skewed drop shadow until you achieve the desired look.

Apply a blur

The Blur filter softens the edges and details of objects. Applying a blur to an object can make it appear as if it is behind other objects, or make an object appear to be in motion.

Text with the Blur filter applied.

1 Select an object to apply a blur to, and select Filters.

2 Click the Add Filter (+) button, and select Blur.

3 Edit the filter settings on the Filter tab:

- To set the width and height of the blur, drag the Blur X and Y slider.
- Select the quality level for the blur. High is approximate to that of a Gaussian blur. Low maximizes playback performance.

Apply a glow

The Glow filter lets you apply a color around the edges of an object.

1 Select an object to apply a glow to, and select Filters.

2 Click the Add Filter (+) button, and select Glow.

3 Edit the filter settings in the Filter tab:

- To set the width and height of the glow, drag the Blur X and Y slider.
- To open the Color Picker and set the glow color, click the color control.
- To set the sharpness of the glow, drag the Strength slider.
- To knock out (or visually hide) the source object and display only the glow on the knockout image, select Knockout.

Using the Glow filter with the Knockout option.

- To apply the glow within the boundaries of the object, select Inner Glow.
- Select the quality level for the glow. High is approximate to that of a Gaussian blur. Low maximizes playback performance.

Apply a bevel

Applying a bevel applies a highlight to the object that makes it appear to be curved up above the background surface.

1 Select an object to apply a bevel to, and select Filters.

2 Click the Add Filter (+) button, and select Bevel.

3 Edit the filter settings in the Filter tab:

- To apply a bevel to the object from the Type pop-up menu, select the bevel type.
- To set the width and height of the bevel, drag the Blur X and Y slider.
- Select a shadow and highlight color for the bevel from the pop-up color palette.
- To set the opacity of the bevel without affecting its width, drag the Strength slider.
- To change the angle of the shadow that the beveled edge casts, drag the Angle dial or enter a value.
- To define the width of the bevel, enter a value for Distance.
- To knock out (or visually hide) the source object and display only the bevel on the knockout image, select Knockout.

Apply a gradient glow

Applying a gradient glow produces a glow look with a gradient color across the surface of the glow. The gradient glow requires one color at the beginning of the gradient with an Alpha value of 0. You cannot move the position of this color, but you can change the color.

1 Select an object to apply a gradient glow to, and select the Filters tab.

2 Click the Add Filter (+) button, and select Gradient Glow.

3 Edit the filter settings on the Filter tab:

- Select the type of glow to apply to the object from the Glow Type pop-up menu.
- To set the width and height of the glow, drag the Blur X and Y slider.
- To set the opacity of the glow without affecting its width, drag the Strength slider.
- To change the angle of the shadow that the glow casts, drag the Angle dial or enter a value.
- To set the distance of the shadow from the object, drag the Distance slider.
- To knock out (or visually hide) the source object and display only the gradient glow on the knockout image, select Knockout.
- Specify a gradient color for the glow. A gradient contains two or more colors that fade or blend into one another. The color you select for the beginning of the gradient is referred to as the *alpha* color.

To change a color in the gradient, select one of the color pointers below the gradient definition bar and click the color space that appears directly below the gradient bar to display the Color Picker. Sliding these pointers adjusts the level and position of that color in the gradient.

To add a pointer to the gradient, click on or below the gradient definition bar. To create a gradient with up to 15 color transitions, add up to 15 color pointers. To reposition a pointer on the gradient, drag the pointer along the gradient definition bar. To remove a pointer, drag it down and off the gradient definition bar.

- Select the quality level for the gradient glow. High is approximate to that of a Gaussian blur. Low maximizes playback performance.

Apply a gradient bevel

Applying a gradient bevel produces a raised look that makes an object appear to be raised above the background, with a gradient color across the surface of the bevel. The gradient bevel requires one color in the middle of the gradient with an alpha value of 0.

1 Select an object to apply a gradient bevel to, and select the Filters tab.

2 Click the Add Filter (+) button, and select Gradient Bevel.

3 Edit the filter settings on the Filter tab:

- Select the type of bevel to apply to the object from the Type pop-up menu.
- To set the width and height of the bevel, drag the Blur X and Y slider.
- To affect the smoothness of the bevel without affecting its width, enter a value for strength.
- To set the angle of the light source, enter a value for Angle, or use the pop-up dial.
- To knock out (or visually hide) the source object and display only the gradient bevel on the knockout image, select Knockout.
- Specify a gradient color for the bevel. A gradient contains two or more colors that fade or blend into one another. The middle pointer controls the alpha color of the gradient. You can change the color of the alpha pointer, but you cannot reposition this color in the gradient.

To change a color in the gradient, select one of the color pointers below the gradient definition bar, and click the color space that appears directly below the gradient bar to display the Color Picker. To adjust the level and position of that color in the gradient, slide these pointers.

To add a pointer to the gradient, click on or below the gradient definition bar. To create a gradient with up to 15 color transitions, add up to 15 color pointers. To reposition a pointer on the gradient, drag the pointer along the gradient definition bar. To remove a pointer, drag it down and off the gradient definition bar.

Apply the Adjust Color filter

To apply only the Brightness control to an object, use the color controls located in the Properties tab of the Property inspector. To provide improved performance to applying a filter, use the Brightness option in the Properties tab.

1 Select an object to adjust the color for, and select the Filters tab.

2 Click the Add Filter (+) button, and select Adjust Color.

3 Drag the slider for the color attributes to adjust, or enter a numerical value in the corresponding box. The attributes and their corresponding values are as follows:

Contrast Adjusts the highlights, shadows, and midtones of an image.

Brightness Adjusts the brightness of an image.

Saturation Adjusts the intensity of a color.

Hue Adjusts the shade of a color.

4 To reset all of the color adjustments to 0 and return the object to its original state, click Reset.

See also

"Change the color and transparency of an instance" on page 213

About blend modes

Blend modes in Flash

Blend modes let you create composite images. *Compositing* is the process of varying the transparency or color interaction of two or more overlapping objects. Blending lets you create unique effects by blending the colors in overlapping movie clips.

A blending mode contains the following elements:

Blend color The color applied to the blend mode.

Opacity The degree of transparency applied to the blend mode.

Base color The color of pixels underneath the blend color.

Result color The result of the blend's effect on the base color.

Blend modes depend on both the color of the object you're applying the blend to and the underlying color. Adobe® recommends that you experiment with the different blend modes to achieve the desired effect.

Normal Applies color normally, with no interaction with the base colors.

Layer Lets you stack movie clips on top of each other without affecting their color.

Darken Replaces only the areas that are lighter than the blend color. Areas darker than the blend color don't change.

Multiply Multiplies the base color by the blend color, resulting in darker colors.

Lighten Replaces only pixels that are darker than the blend color. Areas lighter than the blend color don't change.

Screen Multiplies the inverse of the blend color by the base color, resulting in a bleaching effect.

Overlay Multiplies or screens the colors, depending on the base colors.

Hard Light Multiplies or screens the colors, depending on the blend mode color. The effect is similar to shining a spot light on the object.

Difference Subtracts either the blend color from the base color or the base color from the blend color, depending on which has the greater brightness value. The effect is similar to a color negative.

Add Commonly used to create an animated lightening dissolve effect between two images.

Subtract Commonly used to create an animated darkening dissolve effect between two images.

Invert Inverts the base color.

Alpha Applies an alpha mask.

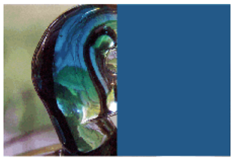
Erase Removes all base color pixels, including those in the background image.

Note: *Erase and Alpha blend modes require that a Layer blend mode be applied to the parent movie clip. You cannot change the background clip to Erase and apply it because the object would appear invisible.*

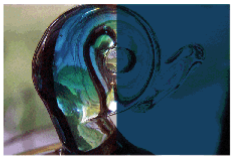
Blend mode examples

The following examples illustrate how different blend modes affect the appearance of an image. The resulting effect of a blend mode might be considerably different, depending on the color of the underlying image and the type of blend mode you apply.

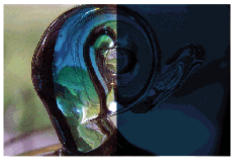
Original image



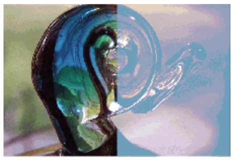
Layer



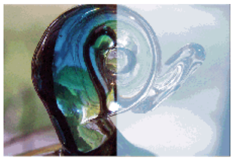
Darken



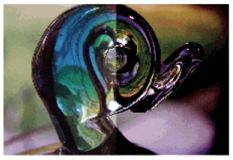
Multiply



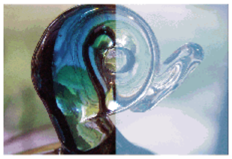
Lighten



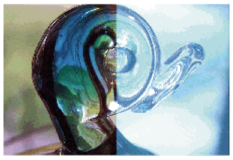
Screen



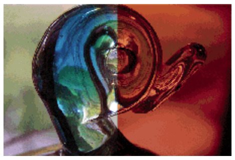
Overlay



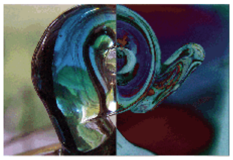
Hard Light



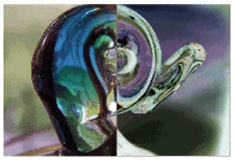
Add



Subtract



Difference



Invert

Apply a blend mode

To apply blends to selected movie clips, use the Property inspector.

Note: *You cannot apply different blend modes to different graphic symbols because Multiple graphic symbols are merged as a single shape when you publish the SWF file.*

See also

"Change the color and transparency of an instance" on page 213

Apply a blend mode to a movie clip

1 Select the movie clip instance (on the Stage) to apply a blend mode to.

2 To adjust the color and transparency of the movie clip instance, use the Color pop-up menu in the Properties panel.

3 Select a blend mode for movie clips from the Blend pop-up menu in the Properties panel. The blend mode is applied to the selected movie clip instance.

4 Verify that the blend mode you selected is appropriate to the effect you're trying to achieve.

Experiment with both the color and transparency settings of the movie clip and the different blend modes to achieve the desired effect.

About importing filters and blends from Fireworks PNG files

When you import Adobe® Fireworks® PNG files, you can retain many of the filters and blending modes applied to objects in Fireworks, and continue to modify those filters and blends using Flash.

Flash only supports modifiable filters and blends for objects imported as text and movie clips. If an effect or blend mode is not supported, Flash rasterizes or ignores it when it is imported. To import a Fireworks PNG file that contains filters or blends that Flash does not support, rasterize the file during the import process. After this operation, you cannot edit the file.

For a video tutorial about the Flash and Fireworks workflow, see www.adobe.com/go/vid0194.

Fireworks effects supported in Flash

Flash imports the following Fireworks effects as modifiable filters:

Fireworks effect	**Flash filter**
Drop shadow	Drop shadow
Solid shadow	Drop shadow
Inner shadow	Drop shadow (with Inner shadow automatically selected)
Blur	Blur (where blurX = blurY=1)
Blur more	Blur (where blurX = blurY=1)
Gaussian blur	Blur
Adjust color brightness	Adjust color
Adjust color contrast	Adjust color

For a video tutorial about the Flash and Fireworks workflow, see www.adobe.com/go/vid0194.

Fireworks blend modes supported in Flash

Flash imports the following Fireworks blend modes as modifiable blends:

Fireworks blending mode	**Flash blending mode**
Normal	Normal
Darken	Darken
Multiply	Multiply
Lighten	Lighten
Screen	Screen

Fireworks blending mode	Flash blending mode
Overlay	Overlay
Hard light	Hard light
Additive	Add
Difference	Difference
Invert	Invert
Alpha	Alpha
Erase	Erase

Flash ignores all other blending modes imported from Fireworks. The blending modes that Flash does not support are Average, Negation, Exclusion, Soft Light, Subtractive, Fuzzy Light, Color Dodge, and Color Burn.

For a video tutorial about the Flash and Fireworks workflow, see www.adobe.com/go/vid0194.

Chapter 12: Working with text

This chapter describes the various ways that text can be incorporated into your Adobe® Flash® CS3 Professional documents.

Text and fonts in Flash

About text

You can include text in your Adobe® Flash® CS3 Professional applications in a variety of ways. You can create text fields containing *static* text, which you create when you author the document. You can also create *dynamic* text fields, which display updating text, such as stock quotes or news headlines, and *input* text fields, which allow users to enter text for forms or surveys.

Flash provides many ways to work with text. For example, you can orient text horizontally or vertically; set attributes such as font, size, style, color, and line spacing; check spelling; transform text by rotating, skewing, or flipping; link text; make text selectable; animate text; control font substitution; and use a font as part of a shared library. Flash documents can use Type 1 PostScript fonts, TrueType, and bitmap fonts (Macintosh only).

You can preserve rich text formatting in text fields, using HTML tags and attributes. When you use HTML text for the content of a dynamic or input text field, the text can flow around an image, such as a SWF or JPEG file or a movie clip. See Using HTML-formatted text in *Learning ActionScript 2.0 in Adobe Flash*.

Like movie clip instances, text field instances are ActionScript™ objects that have properties and methods. By giving a text field an instance name, you can manipulate it with ActionScript. However, you cannot write ActionScript code inside a text instance, because text instances don't have Timelines.

You can use ActionScript to format input and dynamic text, and to create scrolling text fields. ActionScript has events for dynamic and input text fields that you can capture and use to trigger scripts. For information on using ActionScript to control text, see Working with Text and Strings in *Learning ActionScript 2.0 in Adobe Flash*.

For a video tutorial about working with text, see www.adobe.com/go/vid0121.

For a sample file about working with text, see the Flash Samples page at www.adobe.com/go/learn_fl_samples. Download and decompress the Samples zip file and navigate to the Text\TextEnhancements folder to access the sample.

For a video tutorial about using text with Illustrator and Flash, see www.adobe.com/go/vid0199.

See also

"About text fields" on page 264

Flash Text Anti-Aliasing

The Flash text rendering engine that provides clear, high-quality text rendering in Flash (FLA) documents and published SWF files. The Anti-alias for Readability setting makes text more legible, particularly at small font sizes. Custom anti-aliasing lets you specify the thickness and sharpness of fonts used in individual text fields.

High quality anti-aliasing is automatically enabled whenever you publish to Flash Player 8 or later, and Anti-Alias For Readability or Custom Anti-Alias is selected. Anti-Alias For Readability may cause a slight delay when you load Flash SWF files, especially if you are using four or five different character sets in the first frame of a Flash document. High quality anti-aliasing may also increase Flash Player's memory usage. Using four or five fonts, for example, can increase memory usage by approximately 4 MB.

When the publish setting of your file is Adobe® Flash® Player 8 or later, and Anti-Alias For Readability or Custom Anti-Alias is your chosen anti-aliasing option, high-quality anti-aliasing applies to the following:

- Untransformed text that is scaled and rotated

***Note:** Although the text can be scaled or rotated, it must remain flat (that is, untransformed). For example, if you skew the fonts or otherwise manipulate the font shapes, Anti-Alias for Readability is automatically disabled.*

- All font families (including bold, italic, and so on)
- Display sizes of up to 255 points
- Exporting to most non-Flash file formats (GIF or JPEG)

High quality anti-aliasing is disabled under the following conditions:

- Flash Player 7 or earlier is the selected version of Flash Player.
- An anti-aliasing option other than Anti-Alias for Readability or Custom Anti-Alias is selected.
- Text is skewed or flipped.
- The FLA file is exported to a PNG file.

Unicode text encoding in Flash applications

Macromedia Flash Player 7 from Adobe and later supports Unicode text encoding for SWF files in Flash Player format. This support greatly enhances your ability to use multilanguage text in your SWF files, such as two languages within a single text field. Any user with Flash Player 7 or later can view multilanguage text in a Flash Player 7 or later application, regardless of the language used by the operating system running the player.

See also

“Creating multilanguage text” on page 275

Font outlines and device fonts

When you publish or export a Flash document, fonts in text fields are represented by embedded font outlines or by font names. Alternatively, you can use device fonts to specify a general type of font, such as sans serif.

Font outlines and names

For static text, Flash creates outlines of the font and embeds them in the SWF file. Flash Player then uses the outlines to display the text.

For dynamic or input text, Flash stores the font names; Flash Player then locates identical or similar fonts on the user's system when the Flash application is displayed. To ensure that users have the correct fonts for dynamic or input text, you can embed font outlines, but this can increase file size.

Not all fonts displayed in Flash can be exported as outlines with a Flash application. To verify that a font can be exported, use the View > Preview Mode > Antialias Text command to preview the text; jagged type indicates that Flash does not recognize that font's outline and will not export the text.

Device fonts

For static horizontal text, use *device fonts* as an alternative to embedding font outline information. Flash includes three device fonts, _sans (similar to Helvetica or Arial), _serif (similar to Times Roman), and _typewriter (similar to Courier). When you specify one of these fonts and then export the document, Flash Player uses the font on the user's computer that most closely resembles the device font.

Because device fonts are not embedded, they result in a smaller SWF file. They can also result in more legible text at small point sizes (below 10 points). However, if a user's computer does not have an installed font corresponding to the device font, text may look different than expected.

See also

"Create and edit text fields" on page 265

"Use device fonts" on page 274

Masking device fonts

You can use a movie clip to mask device font text in another movie clip. (You cannot mask device fonts by using a mask layer on the Stage.) For this movie clip mask to function, the user must have Macromedia Flash Player 6 (6.0.40.0) from Adobe or later.

When you use a movie clip to mask device font text, Flash uses the rectangular bounding box of the mask as the masking shape. That is, if you create a nonrectangular movie clip mask for device font text in the Flash authoring environment, the mask that appears in the SWF file takes the shape of the rectangular bounding box of the mask, not the shape of the mask itself.

For more information on using a movie clip as a mask, see Using movie clips as masks in *Learning ActionScript 2.0 in Adobe Flash*.

For a sample about device font masking, see the Flash Samples web page at www.adobe.com/go/learn_fl_samples. Download and decompress the Samples zip file and navigate to the Masking\DeviceFontMasking folder to access the sample.

About anti-aliasing text

Anti-aliasing lets you smooth the edges of onscreen text. The anti-aliasing options are particularly effective for rendering smaller font sizes. When anti-aliasing is enabled, all text in the current selection is affected. Anti-aliasing operates with text of all point sizes in the same way.

Anti-aliasing is supported for static, dynamic, and input text if the user has Flash Player 7 or later. It is supported only for static text if the user has an earlier version of Flash Player.

When using small text in a Flash document, keep in mind the following guidelines:

- Sans serif text, such as Helvetica or Arial, appears clearer at small sizes than serif text.
- Some type styles, such as bold and italic, can make text less legible at small sizes.
- In some cases, text appears somewhat smaller than text of the same point size in other applications.

See also

"Work with text anti-aliasing" on page 273

"Font outlines and device fonts" on page 261

Substituting missing fonts

If you work with a document that contains fonts that aren't installed on your system, Flash uses the fonts available on your system. You can select which fonts are substituted for the missing fonts, or you can let Flash substitute missing fonts with the Flash System Default Font.

When a font is substituted, the text is displayed on your system using the substitute font, but the missing font information is saved with the document. If the document is reopened on a system that includes the missing font, the text is displayed in that font.

When you work with a document that includes missing fonts, the missing fonts appear in the font list in the Property inspector. When you select substitute fonts, the substitute fonts names also appear, enclosed by parentheses to identify them as substitute fonts.

If you apply formatting (such as font size or kerning) to the substitute font, check the formatting when the text is displayed in the missing font, because it may not look as expected.

If you install a previously missing font on your system and restart Flash, the font is displayed in documents that use the font, and the font is removed from the Missing Fonts dialog box.

Work with substitute fonts

If you see an alert box indicating missing fonts in a document, you can select substitute fonts in the Font Mapping dialog box.

Specify font substitution

1 When the Missing Fonts alert appears, do one of the following:

- Click Select Substitute Fonts to select substitute fonts from fonts installed on your system and proceed to step 2.
- Click Use Default to use the Flash System Default Font to substitute all missing fonts and to dismiss the Missing Fonts alert. You have finished specifying font substitution.

2 In the Font Mapping dialog box, click a font in the Missing Fonts column to select it. Shift-click to select multiple missing fonts to map them all to the same substitute font.

The default substitute fonts are displayed in the Mapped To column, until you select substitute fonts.

3 Select a font from the Substitute Font pop-up menu.

4 Repeat steps 2 through 3 for all missing fonts.

View all the missing fonts in a document and reselect substitute fonts

1 With the document active in Flash, select Edit > Font Mapping.

2 Click a font in the Missing Fonts column to select it. Shift-click to select multiple missing fonts to map them all to the same substitute font.

The default substitute fonts are displayed in the Mapped To column, until you select substitute fonts.

3 Select a font from the Substitute Font pop-up menu.

4 Repeat steps 2 through 3 for all missing fonts.

View or delete font mappings

1 Close all documents in Flash.

2 Select Edit > Font Mapping.

3 To delete a font mapping, select the mapping and press Delete.

Turn off the Missing Fonts alert

- To turn the alert off for the current document, in the Missing Fonts alert box select Don’t Show Again For This Document, Always Use Substitute Fonts. Select Edit > Font Mapping to view mapping information for the document again.
- To turn the alert off for all documents, select Edit > Preferences (Windows) or Flash > Preferences (Macintosh), and click the Warnings tab. Deselect Warn On Missing Font, and click OK. Select the option again to turn alerts on.

Creating text

About text fields

You can create three types of text fields: static, dynamic, and input. All text fields support Unicode.

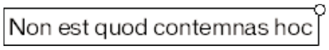
- Static text fields display text that doesn’t change characters dynamically.
- Dynamic text fields display dynamically updating text, such as stock quotes or weather reports.
- Input text fields allow users to enter text in forms or surveys.

You can create horizontal text (with a left-to-right flow) or static vertical text (with either a right-to-left or left-to-right flow).

When creating static text, you can place text on a single line that expands as you type, or in a fixed-width field (for horizontal text) or fixed-height field (for vertical text) that expands and wraps words automatically. When creating dynamic or input text, you can place text on a single line, or create a text field with a fixed width and height.

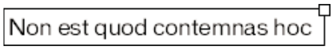
Flash displays a handle on the corner of a text field to identify the type of text field:

- For static horizontal text that extends, a round handle appears at the upper-right corner of the text field.



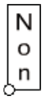
Non est quod contemnas hoc

- For static horizontal text that has a fixed width, a square handle appears at the upper-right corner of the text field.



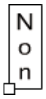
Non est quod contemnas hoc

- For static vertical text that has right-to-left flow and extends, a round handle appears at the lower-left corner of the text field.



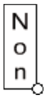
N
o
n

- For static vertical text that has right-to-left flow and a fixed height, a square handle appears at the lower-left corner of the text field.

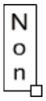


N
o
n

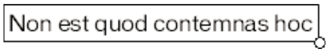
- For static vertical text that has left-to-right flow and extends, a round handle appears at the lower-right corner of the text field.



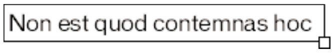
- For static vertical text that has left-to-right flow and a fixed height, a square handle appears at the lower-right corner of the text field.



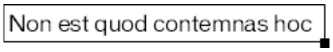
- For dynamic or input text fields that extend, a round handle appears at the lower-right corner of the text field.



- For dynamic or input text that has a defined height and width, a square handle appears at the lower-right corner of the text field.



- For dynamic scrollable text fields, the round or square handle becomes solid black instead of hollow.



Shift-double-click the handle of dynamic and input text fields to create text fields that don't expand when you enter text on the Stage. This allows you to create a text field of a fixed size and fill it with more text than it can display to create scrolling text.

After you use the Text tool to create a text field, use the Property inspector to specify the type of text field, and to set values that control how the text field and its contents appear in the SWF file.

For a video tutorial about working with text, see www.adobe.com/go/vid0121.

For a sample file about working with text, see the Flash Samples page at www.adobe.com/go/learn_fl_samples. Download and decompress the Samples zip file and navigate to the Text\TextEnhancements folder to access the sample.

Create and edit text fields

Text is horizontal by default, however static text can also be aligned vertically.

You can use most common word-processing techniques to edit text in Flash. Use the Cut, Copy, and Paste commands to move text in a Flash file as well as between Flash and other applications.

For a sample file about working with text, see the Flash Samples page at www.adobe.com/go/learn_fl_samples. Download and decompress the Samples zip file and navigate to the Text\TextEnhancements folder to access the sample.

See also

“Setting text attributes” on page 271

Add text to the Stage

1 Select the Text tool T.

2 In the Property inspector (Window > Properties > Properties), select a text type from the pop-up menu to specify the type of text field:

Dynamic Text Creates a field that displays dynamically updating text.

Input Text Creates a field in which users can enter text.

Static Text Creates a field that cannot update dynamically.

3 For static text only: In the Property inspector, click Change Orientation Of Text and select a direction for text orientation and flow. (Horizontal is the default setting.)

4 On the Stage, do one of the following:

- To create a text field that displays text in a single line, click where you want the text to start.
- To create a text field with a fixed width (for horizontal text) or fixed height (for vertical text), position the pointer where you want the text to start and drag to the desired width or height.

***Note:** If you create a text field that extends past the edge of the Stage as you type, the text isn’t lost. To make the handle accessible again, add line breaks, move the text field, or select View > Pasteboard.*

5 Select text attributes in the Property inspector.

Change the size of a text field

❖ Drag the text field’s resize handle.

When text is selected, a blue bounding box lets you resize the text field by dragging one of its handles. Static text fields have four handles that let you resize the text field horizontally. Dynamic text fields have eight handles that let you resize the text field vertically, horizontally, or diagonally.

Switch a text field between fixed-width (or fixed-height) and extending

❖ Double-click a resize handle.

Select characters in a text field

1 Select the Text tool T.

2 Do one of the following:

- Drag to select characters.
- Double-click to select a word.
- Click to specify the beginning of the selection, and Shift-click to specify the end of the selection.
- Press Control+A (Windows) or Command+A (Macintosh) to select all the text in the field.

Select text fields

❖ Using the Selection tool, click a text field. Shift-click to select multiple text fields.

Set dynamic and input text options

1 Click in an existing dynamic text field.

2 In the Property inspector, make sure Dynamic or Input is displayed in the pop-up menu.

3 Enter an instance name for the text field.

4 Specify the height, width, and location of text.

5 Select the font and style.

6 In the Line Type box, specify one of the following options:

Multiline Displays the text in multiple lines.

Single line Displays the text as one line.

Multiline No Wrap Displays text in multiple lines that break only if the last character is a breaking character, such as Enter (Windows) or Return (Macintosh).

7 To enable users to select dynamic text, click Selectable. Deselect this option to prevent users from selecting the dynamic text.

8 To preserve rich text formatting (such as fonts and hyperlinks) with the appropriate HTML tags, click Render Text As HTML.

9 To display a black border and white background for the text field, click Show Border.

10 (Optional) In the Var box, enter the variable name for the text field. (Use this option only when authoring for Macromedia Flash Player 5 from Adobe or earlier.)

Beginning with Macromedia Flash MX (version 6), you assign the text field an instance name using the Property inspector. Although you can use the variable name method with dynamic text fields for backwards compatibility to Macromedia Flash 5 and earlier versions, Adobe doesn't recommend this, because you can't control other text field properties, or apply style sheet settings.

11 Click Embed for embedded font outline options:

Don't embed Specifies that no fonts be embedded.

Auto fill Click Auto Fill to embed all of the characters from the selected text field.

Set preferences for vertical text

1 Select Edit > Preferences (Windows) or Flash > Preferences (Macintosh) and click the Editing tab in the Preferences dialog box.

2 Under Vertical Text, set any of these options:

Default Text Orientation Automatically gives new text fields vertical orientation.

Right to Left Text Flow Makes lines of vertical text fill the page from right to left.

No Kerning Prevents kerning from being applied to vertical text. (Kerning remains enabled for horizontal text.)

Check spelling

You can check spelling in text throughout your Flash document. You can also customize the spell checker.

Use the spell checker

1 Select Text > Check Spelling to view the Check Spelling dialog box.

The box in the upper-left corner identifies words not found in the selected dictionaries, and also identifies the type of element that contains the text (such as a text field or frame label).

2 Do one of the following:

- Click Add To Personal to add the word to your personal dictionary.
- Click Ignore to leave the word unchanged. Click Ignore All to leave all occurrences of the word in the document unchanged.
- Enter a word in the Change To box or select a word from the Suggestions scroll list. Then click Change to change the word or click Change All to change all occurrences of the word in the document.
- Click Delete to delete the word from the document.

3 To end the spelling check, do one of the following:

- Click Close to end spelling before Flash reaches the end of the document.
- Continue checking spelling until you see a notification that Flash has reached the end of the document, then click No to end spelling checking. (Click Yes to continue the spelling check at the beginning of the document.)

Customize the spelling checker

1 Do one of the following:

- Select Text > Spelling Setup. (Use this option if you have not used the Check Spelling feature before.)
- In the Check Spelling dialog box (Text > Check Spelling), click Setup.

2 Set any of the following options:

Document Options Use these options to specify which elements are to be checked.

Dictionaries Lists the built-in dictionaries. You must select at least one dictionary to enable spelling checking.

Personal Dictionary Enter a path or click the folder icon and browse to a document to use as a personal dictionary. (You can modify this dictionary.)

Edit Personal Dictionary Adds words and phrases to your personal dictionary. In the Personal Dictionary dialog box, enter each new item on a separate line in the text field.

Checking Options Use these options to control how Flash handles specific types of words and characters when checking spelling.

Embed and share fonts

You can embed a font in your SWF file so that the font does not need to be present on the devices the SWF file eventually plays back on. To embed a font, create a font library item.

Creating a font library item also allows you to use the font as a shared library item for sharing among multiple SWF files. You must also assign the font item a linkage identifier string and a URL where the document that contains the font symbol will be posted. In this way, you can link to the font and use it in a Flash application. When you use font symbols for dynamic or input text, embed the font outline information.

After a font symbol in a Flash document has an assigned identifier string and URL, use the font symbol in another Flash document by copying the font symbol into the destination FLA file.

See also

“Using shared library assets” on page 218

“Copy library assets between documents” on page 216

Create a font library item

1 Open the library to add a font symbol to.

2 Select New Font from the Library Panel menu.

3 Enter a name for the font item in the Name text field.

4 Select a font from the Font menu or enter the name of a font in the Font text field.

5 (Optional) Select Bold or Italic.

6 (Optional) To embed the font information as bitmap data rather than vector outline data, select the Bitmap Text option, and enter a font size in the Size text field. (Bitmap fonts cannot use anti-aliasing. You must choose Bitmap as the anti-aliasing option in the Property inspector for text that uses this font.)

Note: *The Size setting applies only when you use the Bitmap Text option.*

Assign an identifier string to a font library item

1 Select the font item in the Library panel.

2 Do one of the following:

- Select Linkage from the Library Panel menu.
- Right-click (Windows) or Control-click (Macintosh) the font symbol name in the Library panel, and select Linkage.

3 Under Linkage, select Export for Runtime Sharing.

4 In the Identifier text field, enter a string to identify the font item.

5 In the URL text field, enter the URL of the SWF file that contains the font item.

Scrolling text

There are several ways to create scrolling text in Flash:

- Make dynamic or input text fields scrollable by using menu commands or the text field handle.
- Add a ScrollBar component to a text field to make it scroll. For more information, see “UIScrollBar Component” in the *ActionScript 2.0 Components Language Reference*.
- In ActionScript, use the TextField object’s `scroll` and `maxscroll` properties to control vertical scrolling and the `hscroll` and `maxhscroll` properties to control horizontal scrolling in a text field. See Example: Creating scrolling text in *Learning ActionScript 2.0 in Adobe Flash*.

Make dynamic text scrollable

- Shift-double-click the handle on the dynamic text field.
- Using the Selection tool, select the dynamic text field and then select Text > Scrollable.
- Select the dynamic text field with the Selection tool. Right-click (Windows) or Control-click (Macintosh) the dynamic text field and select Text > Scrollable.

Text effects

You can create text effects by transforming text fields. For example, you can rotate, skew, flip, and scale text fields. (When you scale a text field as an object, the Property inspector does not reflect increases or decreases in point size.) The text in a transformed text field can still be edited, although severe transformations may make it difficult to read.

You can also animate text by using Timeline effects. For example, you can make text bounce, fade in or out, or explode.

See also

“Using Timeline effects” on page 236

Break text apart

You can break apart text to place each character in a separate text field. Then you can quickly distribute the text fields to separate layers and animate each field. (You cannot break apart text in scrollable text fields.)

You can also convert the text to its component lines and fills to reshape, erase, and otherwise manipulate it. As with any other shape, you can individually group these converted characters, or change them to symbols and animate them. After you convert text to lines and fills, you can no longer edit the text.

***Note:** The Break Apart command applies only to outline fonts such as TrueType fonts. Bitmap fonts disappear from the screen when you break them apart. PostScript fonts can be broken apart only on Macintosh systems.*

1 Using the Selection tool ↖, click a text field.

2 Select Modify > Break Apart.

Each character in the selected text is placed in a separate text field. The text remains in the same position on the Stage.

3 Select Modify > Break Apart again to convert the characters to shapes on the Stage.

See also

“Distributing objects to layers for tweened animation” on page 240

“Creating animation” on page 228

Link horizontal text to a URL

1 Select text or a text field:

- Use the Text tool T to select text in a text field.
- To link all the text in a text field, use the Selection tool ↖ to select a text field.

2 In the Link text field 🔗 of the Property inspector (Window > Properties > Properties), enter the URL to which you want to link the text field.

***Note:** To create a link to an e-mail address, use the mailto: URL. For example, enter* **`mailto:adamsmith@example.com`**.

Setting text attributes

About text attributes

You can set the font and paragraph attributes of text. Font attributes include font family, point size, style, color, letter spacing, autokerning, and character position. Paragraph attributes include alignment, margins, indents, and line spacing.

For static text, font outlines are exported in a published SWF file. For horizontal static text, you can use device fonts instead of exporting font outlines.

For dynamic or input text, Flash stores the names of the fonts, and Flash Player locates identical or similar fonts on the user's system. You can also embed font outlines in dynamic or input text fields. Embedding font outlines can increase file size, but it ensures that users have the correct font information.

When creating new text, Flash uses the text attributes that are currently set in the Property inspector. When you select existing text, use the Property inspector to change font or paragraph attributes, and to direct Flash to use device fonts rather than embedding font outline information.

See also

"Font outlines and device fonts" on page 261

"Create and edit text fields" on page 265

Set a font, point size, style, and color

1 Using the Selection tool, select one or more text fields on the Stage.

2 In the Property inspector (Window > Properties > Properties), select a font from the Font pop-up menu, or enter a font name.

Note: *The _sans, _serif, and _typewriter fonts and device fonts can be used only with static horizontal text.*

3 Click the triangle next to the Point Size value and drag the slider to select a value, or enter a font size value.

Font size is set in points, regardless of the current ruler units.

4 To apply bold or italic style, click Bold or Italic.

5 Select a font rendering method from the Anti-Aliasing pop-up menu (directly below the Bold button) to optimize text.

6 To select a fill color for text, click the color control and do one of the following:

- Select a color from the Color Picker.
- Type a color's hexadecimal value in the box in the upper-left corner.
- Click Color Picker and select a color from the system color picker. (When setting the text color, use only solid colors, not gradients. To apply a gradient to text, break the text apart and convert the text to its component lines and fills.)

See also

"Font outlines and device fonts" on page 261

Set letter spacing, kerning, and character position

Letter spacing inserts a uniform amount of space between characters. Use letter spacing to adjust the spacing of selected characters or entire blocks of text.

Kerning controls the spacing between pairs of characters. Many fonts have built-in kerning information. For example, *A* and *V* are often closer together than *A* and *D*. Flash provides horizontal tracking and kerning (for horizontal text) and vertical tracking and kerning (for vertical text).

For vertical text, you can disable kerning by default in Flash Preferences. If you do this and leave the kerning option selected in the Property inspector, kerning is applied to horizontal text only.

1 Using the Text tool T, select one or more sentences, phrases, or text fields on the Stage.

2 In the Property inspector (Window > Properties > Properties), set the following options:

- To specify letter spacing (tracking and kerning), click the triangle next to the Letter Spacing value AV and drag the slider to select a value, or enter a value in the text field.
- To use a font's built-in kerning information, select Auto-Kern.
- To specify character position, click the triangle next to the Character Position option A and select a position from the menu: Normal places text on the baseline, Superscript places text above the baseline (horizontal text) or to the right of the baseline (vertical text), and Subscript places text below the baseline (horizontal text) or to the left of the baseline (vertical text).

Set alignment, margins, indents, and line spacing

Alignment determines the position of each line of text in a paragraph relative to edges of the text field. Horizontal text is aligned relative to the left and right edges of the text field, and vertical text is aligned relative to the top and bottom edges of the text field. Text can be aligned to one edge of the text field, centered in the text field, or aligned to both edges of the text field (full justification).

Margins determine the amount of space between the border of a text field and its text. Indents determine the distance between the margin of a paragraph and the beginning of the first line.

Line spacing determines the distance between adjacent lines in a paragraph. For vertical text, line spacing adjusts the space between vertical columns.

Work with horizontal text

1 Using the Text tool T, select one or more text fields on the Stage.

2 In the Property inspector (Window > Properties > Properties), set the following options:

- To set alignment, click Left, Center, Right, or Full Justification.
- To set the left or right margin, click the Edit Format Options button (shaped like a paragraph symbol, next to the Justify button) to display the Format options dialog box. Click the triangle next to the Left Margin or Right Margin value and drag the slider to select a value, or enter a value in the text field.
- To specify indents, click the Edit Format Options button (shaped like a paragraph symbol, next to the Justify button) to display the Format options dialog box. Click the triangle next to the Indent value and drag the slider, or enter a value in the text field.
- To specify line spacing, click the Edit Format Options button (shaped like a paragraph symbol, next to the Justify button) to display the Format options dialog box. Click the triangle next to the Line Spacing value and drag the slider to select a value, or enter a value in the text field.

Work with vertical text

1 Using the Text tool T, select one or more text fields on the Stage.

2 To apply settings to existing text, select text fields on the Stage.

3 In the Property inspector (Window > Properties > Properties), set the following options:

- To set alignment, click Top, Center, Bottom, or Full Justification.
- To set the top or bottom margin, click the Edit Format Options button (shaped like a paragraph symbol, next to the Justify button) to display the Format options dialog box. Click the triangle next to the Top Margin or the Bottom Margin value and drag the slider to select a value, or enter a value in the text field.
- To specify indents, click the Edit Format Options button (shaped like a paragraph symbol, next to the Justify button) to display the Format options dialog box. Click the triangle next to the Indent value and drag the slider to select a value, or enter a value in the text field. If the text flows left to right, the indent moves the leftmost line down; if the text flows right to left, the indent moves the rightmost line down.
- To specify line spacing, click the Edit Format Options button (shaped like a paragraph symbol, next to the Justify button) to display the Format options dialog box. Click the triangle next to the Line Spacing value and drag the slider to select a value, or enter a value in the text field.

Work with text anti-aliasing

Flash provides improved font rasterization that lets you specify the anti-aliasing properties for fonts. The improved anti-aliasing capabilities are available only for SWF files published for Flash Player 8 or later. If you are publishing files for earlier versions of Flash Player, you can only use the Anti-Alias For Animation feature.

Apply anti-aliasing for each text field rather than each character. Also, when you open existing FLA files in Flash 8 or later, the text is not automatically updated to the advanced anti-aliasing options; you must select individual text fields and manually change the anti-aliasing settings.

When you open a FLA file created for use with Flash Player 7 or earlier, the text Property inspector sets the anti-alias option to its equivalent anti-aliasing option from Flash MX 2004. However, to use the advanced anti-aliasing features, you must upgrade the FLA content for Flash Player 8 or later.

Choose an anti-aliasing option for selected text

❖ In the Property inspector, choose one of the following options from the Anti-Aliasing pop-up menu:

Use Device Fonts Specifies that the SWF file use the fonts installed on the local computer to display the fonts. Typically, device fonts are legible at most font sizes. Although this option doesn’t increase the size of the SWF file, it forces you to rely on the fonts installed on the user’s computer for font display. When using device fonts, choose only commonly installed font families.

Bitmap Text (No Anti-Alias) Turns off anti-aliasing and provides no text smoothing. The text is displayed using sharp edges, and the resulting SWF file size is increased because the font outlines are embedded in the file. Bitmap text is sharp at the exported size, but scales poorly.

Anti-Alias For Animation Creates a smoother animation by ignoring alignment and kerning information. This option creates a larger SWF file, because font outlines are embedded. For legibility, use 10-point or larger type when specifying this option.

Anti-Alias For Readability Uses the Flash text rendering engine to improve the legibility of fonts, particularly at small sizes. This option creates a larger SWF file, because font outlines are embedded. To use this option, you must publish to Flash Player 8 or later. (Do not use this option if you intend to animate text; instead, use Anti-Alias For Animation.)

Custom Anti-Alias Lets you modify the font’s properties. Use Sharpness to specify the smoothness of the transition between the text edges and the background. Use Thickness to specify how thick the font anti-aliasing transition appears. (Larger values cause the characters to look thicker.) Specifying Custom Anti-Alias creates a larger SWF file, because font outlines are embedded. To use this option, you must publish to Flash Player 8 or later.

Upgrade content for Flash 8 or later anti-aliasing

1 Open a FLA file created for use with Macromedia Flash Player 7 or earlier.

2 In the Publish Settings dialog box (File > Publish Settings), select Flash Player 8 or Flash Player 9 from the Version pop-up menu.

3 Select the text field to apply the Anti-Alias For Readability or Custom Anti-Alias option to.

4 In the Property inspector, select Anti-Alias For Readability or Custom Anti-Alias from the Font Rendering Method pop-up menu.

Make text selectable

Static horizontal text or dynamic text can be selectable by users viewing your Flash application. (Input text is selectable by default.) After selecting text, the user can copy, cut, and then paste the text into a new document.

1 Using the Text tool T, select the horizontal text that you want to make selectable.

2 In the Property inspector (Window > Properties > Properties), select Static Text or Dynamic Text.

3 Click Selectable Ab.

Use device fonts

Use device fonts in static horizontal text only.

1 Using the Selection tool, select one or more text fields.

2 In the Property inspector (Window > Properties > Properties), select Static Text from the pop-up menu.

3 In the Font pop-up menu, select a device font:

_sans a font similar to Helvetica or Arial.

_serif A font similar to Times Roman.

_typewriter A font similar to Courier.

See also

“Font outlines and device fonts” on page 261

“Masking device fonts” on page 262

Chapter 13: Creating multilanguage text

Adobe® Flash® CS3 Professional provides several features that enhance the work flow for authoring multiple language Unicode-based applications.

Creating multilanguage text

About multilanguage text

You can include multilanguage text in your document in the following ways:

- The Strings panel lets localizers edit strings in a central location in Flash or in external XML files with their preferred software or translation memory. Flash supports multiline strings in both the Strings panel and the XML files.
- Select which character sets to embed in your applications, which limits the number of character glyphs in your published SWF file and reduces its size.
- Use a Western-style keyboard to create text on the Stage in Chinese, Japanese, and Korean.
- If you have Unicode fonts installed on your system, enter text directly into a text field. Because the fonts are not embedded, your users must also have Unicode fonts.

Other, less common methods of including multilingual text in your Flash documents include the following:

- Include an external text file in a dynamic or input text field by using the `#include` action.
- Load external text or XML files into a Flash application at runtime by using the `loadVariables` or `getURL` actions, or the `LoadVars` or `XML` objects.
- Enter Unicode escape characters in the string value for a dynamic or input text field variable.
- Create an embedded font as a symbol in your Library.

For Unicode-encoded text to appear correctly, users must have access to fonts containing the glyphs (characters) used in that text.

For a sample of multilingual content, see the Flash Samples page at www.adobe.com/go/learn_fl_samples. Download and decompress the Samples zip file and navigate to the Text\MultilingualContent folder to access the sample.

See also

“Workflow for authoring multilanguage text with the Strings panel” on page 281

“Multilanguage text and ActionScript” on page 288

“Create multilanguage documents using the #include action” on page 288

“Use ActionScript to load external files” on page 288

“Creating multilanguage documents by using text variables” on page 289

“Non-Unicode external files” on page 279

“Embed and share fonts” on page 268

About fonts for Unicode-encoded text

When you use external files that are Unicode encoded, your users must have access to fonts containing all the glyphs used in your text files. By default, Flash stores the names of fonts used in dynamic or input text files. During SWF file playback, Flash Player 7 (and earlier versions) looks for those fonts on the operating system running the player.

If the text in a SWF file contains glyphs that the specified font does not support, both Flash Player 7 and Flash Player 8 attempt to locate a font on the user's system that supports those glyphs. The player cannot always locate an appropriate font. This behavior depends on the fonts available on the user's system, as well as on the operating system running Flash Player.

Embedded fonts for dynamic or input text fields

For dynamic or input text fields, embed fonts. However, some fonts, particularly those used for Asian languages, can add significantly to the SWF file size when embedded. With Flash, you can select ranges of fonts to embed.

You can also embed fonts by creating a font symbol in the library.

See also

"Embed and share fonts" on page 268

Select and embed a range of fonts

1 On the Stage, select a dynamic or input text field, show the Property inspector (Window > Properties > Properties), and click Embed.

2 Select the range of font sets to embed by doing any of the following:

- Click a character set in the Character Set text field.
- Select multiple ranges by Shift-clicking the first and last fonts of a contiguous range of fonts, or by Control-clicking (Windows) or Command-clicking (Macintosh) to select noncontiguous fonts, click Auto Fill, and then click OK.

***Note:** Select only the font sets to embed, so you do not exceed the internal maximum number of glyphs for the authoring tool (approximately 65.500). Flash does not perform error-checking to confirm that the selected character set contains glyphs. During the publish or export procedure, only glyphs that are present in the font are embedded in the SWF file.*

Embed font sets from text on the Stage

1 On the Stage, select a dynamic or input text field, show the Property inspector (Window > Properties > Properties), and click Embed.

2 Select the character sets to embed and click AutoFill.

Remove embedded font sets

1 On the Stage, select a dynamic or input text field, show the Property inspector (Window > Properties > Properties), and click Embed.

2 Click Don't Embed to remove all of the embedded fonts.

3 Click OK to close the dialog box, or select other fonts sets to embed in the SWF file.

XML font embedding table

When you select ranges of fonts to embed, the list of selected fonts is stored and maintained as an external XML file and resides in the user configuration folder. It is named Unicode_Table.xml and contains the one-to-many relationship between a particular language and all the necessary Unicode glyph ranges as shown in the following Korean examples.

The font set groupings are based on the Unicode Blocks as defined by the Unicode Consortium. To provide a simpler workflow, when you select a particular language, all related glyph ranges are embedded even if they are scattered into disjointed groupings.

For example, if you select Korean, the following Unicode character ranges are embedded:

3131-318E Hangul symbols

3200-321C Hangul specials

3260-327B Hangul specials

327F-327F Korean symbol

AC00-D7A3 Hangul symbols

If you select Korean + CJK, a larger font set is embedded:

3131-318E Hangul symbols

3200-321C Hangul specials

3260-327B Hangul specials

327F-327F Korean symbol

4E00-9FA5 CJK symbols

AC00-D7A3 Hangul symbols

F900-FA2D CJK symbols

The following table gives more details about the font selections for embedded fonts:

Range	Description
Uppercase [A–Z]	Basic Latin uppercase glyphs
Lowercase [a–z]	Basic Latin lowercase glyphs
Numerals [0–9]	Basic Latin numeral glyphs
Punctuation [!@#%...]	Basic Latin punctuation
Basic Latin	Basic Latin glyphs within the Unicode range 0x0021 to 0x007E
Japanese Kana	Hiragana and Katakana glyphs (including half-width forms)
Japanese Kanji – Level 1	Japanese Kanji characters
Japanese (All)	Japanese Kana and Kanji (including punctuation and special characters)

Range	Description
Basic Hangul	Most commonly used Korean characters, Roman characters, punctuations, and special characters/symbols
Hangul (All)	11,720 Korean characters (sorted by Hangul syllables), Roman characters, punctuations, and special characters/symbols)
Traditional Chinese – Level 1	5000 most commonly used Traditional Chinese characters used in Taiwan
Traditional Chinese (All)	All Traditional Chinese characters used in Taiwan and Hong Kong, and punctuations
Simplified Chinese – Level 1	6000 most commonly used Simplified Chinese characters used in mainland of China and punctuations
Chinese (All)	All Traditional and Simplified Chinese characters and punctuations
Thai	All Thai glyphs
Devanagari	All Devanagari glyphs
Latin I	Latin-1 Supplement range 0x00A1 to 0x00FF (including punctuation, superscripts and subscripts, currency symbols, and letter-like symbols)
Latin Extended A	Latin Extended-A range 0x0100 to 0x01FF (including punctuation, superscripts and subscripts, currency symbols, and letter-like symbols)
Latin Extended B	Latin Extended-B range 0x0180 to 0x024F (including punctuation, superscripts and subscripts, currency symbols, and letter-like symbols)
Latin Extended Add'l	Latin Extended Additional range 0x1E00 to 0x1EFF (including punctuation, superscripts and subscripts, currency symbols, and letter-like symbols)
Greek	Greek and Coptic, plus Greek Extended (including punctuation, superscripts and subscripts, currency symbols, and letter-like symbols)
Cyrillic	Cyrillic (including punctuation, superscripts and subscripts, currency symbols, and letter-like symbols)
Armenian	Armenian plus ligatures
Arabic	Arabic plus Presentation Forms-A and Presentation Forms-B
Hebrew	Hebrew plus Presentation Forms (including punctuation, superscripts and subscripts, currency symbols, and letter-like symbols)

Non-Unicode external files

If you load external text or XML files that are not Unicode-encoded into a Flash Player 7 application, the text in the external files does not appear correctly when Flash Player attempts to show them as Unicode. To tell Flash Player to use the traditional code page of the operating system that is running the player, add the following code as the first line of code in the first frame of the Flash application that is loading the data:

```
system.useCodepage = true;
```

Set the `system.useCodepage` property only once in a document; do not use it multiple times in a document to make the player interpret some external files as Unicode and some as other encoding, because this can yield unexpected results.

If you set the `system.useCodepage` property to `true`, the traditional code page of the operating system running the player must include the glyphs used in your external text file for the text to appear. For example, if you load an external text file that contains Chinese characters, those characters do not appear on a system that uses the CP1252 code page, because that code page does not include Chinese characters. To ensure that users on all platforms can view external text files used in your Flash applications, encode all external text files as Unicode and leave the `system.useCodepage` property set to `false` by default. This causes Flash Player to interpret the text as Unicode. For more information, see useCodepage (System.useCodepage property) in the *ActionScript 2.0 Language Reference*.

Enter Asian characters on a Western keyboard

With Flash, you can use Input Method Editors (IMEs) and standard Western keyboards to enter Asian characters on the Stage. Flash supports more than two dozen IMEs.

For example, to create a website that reaches a broad range of Asian viewers, you can use a standard Western (QWERTY) keyboard and change the IME to create text in Chinese, Japanese, and Korean.

***Note:** This feature affects only text input on the Stage, not text entered in the Actions panel. This feature is available for all supported Windows operating systems and Mac OS X.*

1 Select Edit > Preferences (Windows) or Flash > Preferences (Macintosh), and click Text in the Category list.

2 Under Input Method, select one of the options to input characters from a Western keyboard. The default is Chinese and Japanese and it should also be selected for Western languages.

Encoding text formats

Text encoding

All text in a computer is encoded as a series of bytes. Many different forms of encoding (and therefore, different bytes) represent text. Different kinds of operating systems use different kinds of encoding for text. For example, Western Windows operating systems usually use CP1252 encoding; Western Macintosh operating systems usually use MacRoman encoding; Japanese Windows and Macintosh systems usually use Unicode encoding.

Unicode can encode most languages and characters used throughout the world. The other forms of text encoding that computers use are subsets of the Unicode format, tailored to specific regions of the world. Some of these forms are compatible in some areas and incompatible in other areas, so using the correct encoding is critical.

Unicode has several forms. Flash Player versions 6 and 7 and later support text or external files in the 8-bit Unicode format UTF-8, and in the 16-bit Unicode formats UTF-16 BE (Big Endian) and UTF-16 LE (Little Endian).

Unicode and Flash Player

Flash Player 6 and later versions support Unicode text encoding. Users with Flash Player 6 or later can view multi-language text, regardless of the language that the operating system running the player uses, if they have the correct fonts installed.

Flash Player assumes that all external text files associated with a Flash Player application are Unicode encoded, unless you tell the player otherwise.

For Flash applications in Flash Player 5 or earlier that are authored in Flash MX or earlier, Flash Player 6 and earlier versions display the text by using the traditional code page of the operating system running the player.

For background information on Unicode, see Unicode.org.

Text encoding in Flash Player

By default, Flash Player 7 and later assumes that all text it encounters is Unicode encoded. If your document loads external text or XML files, the text in these files should be UTF-8 encoded. Create these files by using the Strings panel or using a text or HTML editor that can save the files in Unicode format.

Unicode encoding formats that Flash Player supports

When reading text data in Flash, Flash Player looks at the first two bytes in the file to detect a byte order mark (BOM), a standard formatting convention used to identify the Unicode encoding format. If no BOM is detected, the text encoding is interpreted as UTF-8 (an 8-bit encoding format). It is recommended that you use UTF-8 encoding in your applications.

If Flash Player detects either of the following BOMs, the text encoding format is interpreted as follows:

- If the first byte of the file is OxFE and the second is OxFF, the encoding is interpreted as UTF-16 BE (Big Endian). This is used for Macintosh operating systems.
- If the first byte of the file is OxFF and the second is OxFE, the encoding is interpreted as UTF-16 LE (Little Endian). This is used for Windows operating systems.

Most text editors that can save files in UTF-16BE or LE automatically add the BOMs to the files.

***Note:** If you set the `system.useCodepage` property to `true`, the text is interpreted using the traditional code page of the operating system that is running the player; it is not interpreted as Unicode.*

Encoding in external XML files

You cannot change the encoding of an XML file by changing the encoding tag. Flash Player identifies the encoding of an external XML file using the same rules as for all external files. If no BOM is encountered at the beginning of the file, the file is assumed to be in UTF-8 encoding. If a BOM is encountered, the file is interpreted as UTF-16BE or LE.

See also

"Non-Unicode external files" on page 279

Authoring multilanguage text

Workflow for authoring multilanguage text with the Strings panel

The Strings panel lets you create and update multilingual content. You can specify content for text fields that span multiple languages, and have Flash automatically determine the content that should appear in a certain language based on the language of the computer running Flash Player.

The following steps describe the general work flow:

1. Author a FLA file in one language.

Any text to enter in another language must be in a dynamic or input text field.

2. In the Strings Panel Settings dialog box, select the languages to include and designate one of them as the default language.

A column for the language is added to the Strings panel. When you save, test, or publish the application, a folder with an XML file is created for each language.

3. In the Strings panel, encode each text string with an ID.

4. Publish the application.

A folder is created for each language you select, and within each language folder is an XML file for that language.

5. Send the published FLA file and XML folders and files to your translators.

Author in your native language and let the translators make the translation. They can use translation software directly in the XML files or in the FLA file.

6. When you receive the translations from your translators, import the translated XML files back into the FLA file.

Select and remove languages for translation

As many as 100 languages can appear on the Stage and in the Strings panel for translation. Each language you select becomes a column in the Strings panel. To show the text on the Stage in any of the languages you selected, change the Stage language. The selected language appears when you publish or test the file.

When selecting languages, use any of the languages provided in the menu, as well as any other Unicode-supported language.

See also

“Publishing overview” on page 418

Select a language

1 Select Window > Other Panels > Strings, and click Settings.

2 Add a language by doing one of the following:

- In the Languages box, highlight a language to select, and click Add.

- If the language does not appear in the Languages box, in the blank field below the Languages box, type a language code in the format *xx.* (The language code is from ISO 639-1.) Click Add.

3 Repeat step 3 until you have added all the necessary languages.

4 Select a default language from the Default runtime language menu. This language appears on systems that do not have one of the active languages you selected.

5 To load an XML file for the languages from a different URL at runtime, type the URL in the URL text field and click OK.

A column for each selected language appears in the Strings panel. The columns appear in alphabetical order.

6 Save the FLA file. When you save the FLA file, a folder for each language you selected is created in the same folder indicated in the SWF publish path. If no SWF publish path has been selected, it is created in the folder the FLA file resides in. Within each language file an XML file is created that is used to load translated text.

Remove a language

1 Select Window > Other Panels > Strings, and click Settings.

2 In the Active languages field, highlight a language and click Remove.

3 Repeat step 3 until you have removed all the unwanted languages.

4 When you finish removing languages, click OK.

The column for each removed language no longer appears in the Strings panel.

***Note:** When you remove a language from the Strings panel, the language XML file is not deleted from the local file system. This lets you add the language back into the Strings panel by using the previous XML file, and prevents accidental deletion. To completely remove the language, you must delete or replace the language XML file.*

Add strings to the Strings panel

Assign text strings to the Strings panel in the following ways:

- Assign a string ID to a dynamic or input text field
- Add a string to the Strings panel without assigning it to a text field
- Assign an existing string ID to an existing dynamic or input text field

See also

“Creating text” on page 264

Assign a string ID to a text field

1 Select Window > Other Panels > Strings.

2 Select the Text tool. On the Stage, create an input or dynamic text field.

3 While the text field is selected, type a unique ID in the ID field in the Strings panel.

4 Click the Settings button and select a language or languages from the list in the Settings dialog box. The languages you select should include the default language you wish to use and any other languages in which you plan to publish your work.

5 Click Apply.

***Note:** If a static text field is selected on the Stage, the Stage text selection section on the Strings panel displays the message "Static text cannot have an ID associated with it." If a nontext item is selected or multiple items are selected, the message "Current selection cannot have an ID associated with it" appears.*

Add a string ID to the Strings panel without assigning it to a text field

1 Select Window > Other Panels > Strings.

2 Click the Settings button and select a language or languages from the list in the Settings dialog box. The languages you select should include the default language you wish to use and any other languages in which you plan to publish your work.

3 Type a new string ID and new string in the Strings panel, and click Apply.

Assign an existing ID to a text field

1 Select the Text tool. On the Stage, create an input or dynamic text field.

2 Type the name of an existing ID in the ID section of the Strings panel, and click Apply.

***Note:** Press Shift+Enter to apply the ID to the text field, or Enter if the focus is on the ID field.*

Change the language displayed on the Stage

1 Select Window > Other Panels > Strings.

2 In the Stage Language menu, select the language to use for the Stage language. This must be a language you added as an available language.

After you change the Stage language, any new text you type on the Stage appears in that language. If you previously entered text strings for the language in the Strings panel, any text on the Stage appears in the selected language. If not, the text fields already on the Stage are blank.

Editing strings in the Strings panel

After you enter text strings in the Strings panel, use one of the following methods to edit the text strings:

- Directly in the Strings panel cells.
- On the Stage in the language selected as the Stage language, using features such as find and replace and spelling checking. Text that you change using these features is changed on the Stage and in the Strings panel.
- Edit the XML file directly.

See also

"Translate text in the Strings panel or an XML file" on page 287

"Find and Replace" on page 76

"Check spelling" on page 267

Publishing multilanguage FLA files

When you save, publish, or test the FLA file, a folder with an XML file is created for each available language you selected in the Strings panel. The default location for the XML folders and files is the same folder indicated as the SWF publish path. If no SWF publish path was selected, the XML folder and files are saved in the folder in which the FLA file is located. For example, if you have a file named Test in the mystuff folder, and you selected English (en), German (de), and Spanish (es) as active languages, and you did not select a SWF publish path, when you save the FLA file, the following folder structure is created:

```
\mystuff\Test.fla
\mystuff\de\Test_de.xml
\mystuff\en\Test_en.xml
\mystuff\es\Test_es.xml
```

When you start a SWF file, you also need to start the associated XML files with the string translations in the web server. The first frame that contains text cannot appear until the entire XML file is downloaded.

See also

"Publishing Flash documents" on page 418

Manually replace strings at publish time

Manually replace strings by using the Stage language when you publish your Flash SWF file. This method uses the Stage language to replace all instances of input and dynamic text with an associated string ID. In this case, text strings are only updated when you publish the SWF file; language detection is not automatic, and you must publish a SWF file for each language to support.

1 Select Window > Other Panels > Strings, and click Settings.

2 Select the Replace Strings Automatically At Runtime check box.

Use automatic language detection with the default language

You can change the default runtime language to any language that you selected as an available language. When automatic language detection is on, and you view the SWF file on the system that uses the language, any system that is set to a language other than one of the active languages uses the default language. For example, if you set your default language to English and you select ja, en, and fr as active languages, users who have their system language set to Japanese, English, or French automatically see text strings in their chosen language. However, users who have their system language set to Swedish, which is not one of the selected languages, automatically see text strings in the default language you selected—in this case, English.

1 Select Window > Other Panels > Strings, and click Settings.

2 In the Default language menu, select the default language. This must be a language you added as an available language.

3 To enable automatic language detection, select Replace Strings Automatically At Runtime, and click OK.

4 Flash generates the following ActionScript™, which stores the language XML file paths. Use this code as a starting point for your own language detection script.

```
import mx.lang.Locale;
Locale.setFlaName("<flaFileName>");
Locale.setDefaultLang("langcode");
Locale.addXMLPath("langcode", "url/langcode/flaname_langcode.xml");
```

***Note:** The ActionScript code that the Strings panel generates does not use the `Locale.initialize` function. Decide how to call this function based on the language detection customizations your project requires.*

Use custom language detection

To access the language XML files to control text replacement at a time that you designate, create your own custom component or use ActionScript code. For example, you might create a pop-up menu that lets users select a language for viewing content.

For information on writing ActionScript code to create custom language detection, see About the Strings panel in *Learning ActionScript 2.0 in Adobe Flash*.

1 Select Window > Other Panels > Strings, and click Settings.

2 In the Default Language menu, select the default language.

This must be a language you added as an available language.

3 Select the Replace Strings Via ActionScript check box, and click OK.

Flash generates the following ActionScript code, which stores the language XML file paths. Use this code as a starting point for your own language detection script.

```
import mx.lang.Locale;
Locale.setFlaName("<flaFileName>");
Locale.setDefaultLang("langcode");
Locale.addXMLPath("langcode", "url/langcode/flaname_langcode.xml");
```

***Note:** The ActionScript that the Strings panel generates does not use the `Locale.initialize` function. Decide how to call this function based on the language detection customizations your project requires.*

XML file format

About the XML file format

Exported XML is in UTF-8 format and follows the XML Localization Interchange File Format (XLIFF)1.0 standard. It defines a specification for an extensible localization interchange format that lets any software provider produce a single interchange format that can be delivered to, and understood by, any localization service provider. For more information about XLIFF, see www.oasis-open.org/committees/xliff/.

XLIFF examples

If any of the following characters are entered in the Strings panel, they are replaced by the appropriate entity reference when written to XML files:

Character	Replaced by
&	&amp;
'	&apos;
"	&quot;
<	&lt;
>	&gt;

Exported XML file sample

The following examples show what an XML file that the Strings panel generates looks like in the source language—in this example, English—and in another language—in this example, French:

English source version sample

```
<?xml version="1.0" encoding="UTF-8"?>
<!DOCTYPE xliff PUBLIC "-//XLIFF//DTD XLIFF//EN"
"http://www.oasis-open.org/committees/xliff/documents/xliff.dtd" >
<xliff version="1.0" xml:lang="en">
<file datatype="plaintext" original="MultiLingualContent.fla" source-language="EN">
        <header></header>
        <body>
            <trans-unit id="001" resname="IDS_GREETINGS">
                <source>welcome to our web site!</source>
            </trans-unit>
            <trans-unit id="002" resname="IDS_MAILING LIST">
                <source>Would you like to be on our mailing list?</source>
            </trans-unit>
            <trans-unit id="003" resname="IDS_SEE YOU">
                <source>see you soon!</source>
            </trans-unit>
            <trans-unit id="004" resname="IDS_TEST">
                <source></source>
            </trans-unit>
        </body>
    </file>
</xliff>
```

French version sample

```
<?xml version="1.0" encoding="UTF-8"?>
<!DOCTYPE xliff PUBLIC "-//XLIFF//DTD XLIFF//EN"
"http://www.oasis-open.org/committees/xliff/documents/xliff.dtd" >
<xliff version="1.0" xml:lang="fr">
<file datatype="plaintext" original="MultiLingualContent.fla" source-language="EN">
        <header></header>
        <body>
            <trans-unit id="001" resname="IDS_GREETINGS">
                <source>Bienvenue sur notre site web!</source>
            </trans-unit>
            <trans-unit id="002" resname="IDS_MAILING LIST">
                <source>Voudriez-vous être sur notre liste de diffusion?</source>
            </trans-unit>
            <trans-unit id="003" resname="IDS_SEE YOU">
                <source>A bientôt!</source>
            </trans-unit>
            <trans-unit id="004" resname="IDS_TEST">
                <source></source>
            </trans-unit>
        </body>
    </file>
</xliff>
```

Translate text in the Strings panel or an XML file

When sending files to translators, include not only the FLA file but also the folders for the XML files and the XML file for each language.

Translators can either work directly in the language columns in the Strings panel or work in the XML files for each language to translate the FLA file to selected languages. If you translate directly in the XML file, you must either import the XML file to the Strings panel or save it in the default directory for that language.

Translate text in the Strings panel

1 Select Window > Other Panels > Strings.

2 For each language to be translated, select the appropriate language column, then type the translated text for that language to be associated with each string ID.

3 To show the text on the Stage in the language you selected, select the language in the Stage Language field.

4 When you are finished, save, publish, or test the file.

All XML files for all languages are overwritten with the information in the Strings panel.

Note: *To preserve the translation in an XML file, save it in a different folder.*

Translate text in an XML file

1 Using an XML file editor or translating software, open the folder for the desired language, then the XML file for that language. The XML file is populated with the IDs for each text string.

2 Enter the text string for the language next to the ID.

3 If necessary, import the translated XML file into the Strings panel.

Import an XML file into the Strings panel

After you modify an XML file, if you place it in the folder specified in the Strings panel for that language, the XML file is loaded into the Flash document (FLA file) when it opens.

Regardless of where the XML file you imported was located, when you save, test, or publish the FLA file, a folder for each language in the Strings panel and an XML file for each language are created in the location indicated for publishing SWF files. If no publish path is indicated, the folder and file are saved in the same folder in which the FLA file is located. The XML files that the Strings panel generates are always populated with the information in the Strings panel.

Alternatively, import an XML file into the Strings panel from another location. After you import it, when you save, test, or publish the file, the XML file in the folder specified for that language is overwritten. You cannot import an XML file for a language unless it is already selected as an available language in the Strings panel. You can also add a language and import an XML file with the translation for that language.

1 Select Window > Other Panels > Strings, and click Import XML.

2 In the Select a Language menu, select the language of the XML file you are importing, and click OK.

3 Navigate to the folder and XML file to import.

The XML information is loaded into the column in the Strings panel for the language you selected in step 3.

Note: *Select the same language in steps 2 and 3. Otherwise, you could, for example, import a French XML file into the column for German.*

See also

“Publishing overview” on page 418

Multilanguage text and ActionScript

Use ActionScript to load external files

To load existing XML data, or use a different format for the XML file, use the `loadVariables` action, the `getURL` action, the `LoadVars` object, or the `XML` object to create a document that contains multilanguage text by placing the text in an external text or XML file and loading the file into the movie clip at runtime.

Save the external file in UTF-8 (recommended), UTF-16BE, or UTF-16LE format, using an application that supports the format. If you are using UTF-16BE or UTF-16LE format, the file must begin with a BOM to identify the encoding format to Flash Player. The following table lists the BOM to include to identify the encoding:

***Note:** Most text editors that can save files in UTF-16BE or LE automatically add the BOMs to the files.*

UTF Format	First Byte	Second Byte
UTF-16BE	OxFE	OxFF
UTF-16LE	OxFF	OxFE

***Note:** If the external file is an XML file, you cannot use an XML encoding tag to change the file encoding. Save the file in a supported Unicode format.*

1 In the Flash authoring application, create a dynamic or input text field to show the text in the document.

2 In the Property inspector, with the text field selected, assign an instance name to the text field.

3 Outside of Flash, create a text or XML file that defines the value for the text field variable.

4 Save the XML file in UTF-8 (recommended), UTF-16BE, or UTF-16LE format.

5 Use one of the following ActionScript procedures to reference the external file and load it into the dynamic or input text field:

- Use the `loadVariables` action to load an external file.
- Use the `getURL` action to load an external file from a specified URL.
- Use the `LoadVars` object (a predefined client-server object) to load an external text file from a specified URL.
- Use the `XML` object (a predefined client-server object) to load an external XML file from a specified URL. For more information, see XML in the *ActionScript 2.0 Language Reference*.

See also

“Using the XMLConnector component to connect to external XML files” on page 290

“Unicode and Flash Player” on page 280

“Working with text” on page 260

Create multilanguage documents using the #include action

To create a document that contains multiple languages, use the `#include` action.

Use an application that supports UTF-8 encoding, such as Dreamweaver, to save the text file in UTF-8 format.

To identify the file as Unicode to the Flash authoring tool, include the following header as the first line of the file:

```
//!-- UTF8
```

Note: *Include a space after the second dash (-).*

By default, the Flash authoring application assumes that external files that use the `#include` action are encoded in the traditional code page of the operating system running the authoring tool. Using the `//!-- UTF8` header in a file tells the authoring tool that the external file is encoded as UTF-8.

1 In the Flash authoring tool, create a dynamic or input text field to display the text in the document.

2 In the Property inspector, with the text field selected, assign an instance name to the text field.

3 Outside of Flash, create a text file that defines the value for the text field variable. Add the **`//!-- UTF8`** header at the beginning of the file.

4 Save the text file in UTF-8 format.

5 To include the external file in the dynamic or input text field, use the `#include` directive. For more information, see `#include` directive in the *ActionScript 2.0 Language Reference*.

See also

"Working with text" on page 260

Creating multilanguage documents by using text variables

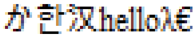
To include Unicode-encoded contents in text variables, use the syntax `\uXXXX`, where `XXXX` is the four-digit hexadecimal code point, or *escape* character, for the Unicode character. The Flash authoring tool supports Unicode escape characters through `\uFFFF`. To find the code points for Unicode characters, see the Unicode Standard at Unicode.org.

You can use Unicode escape characters only in text field variables. You cannot include Unicode escape characters in external text or XML files; Flash Player 6 does not recognize Unicode escape characters in external files.

For example, to set a dynamic text field (with the `myTextVar` instance name) that contains Japanese, Korean, Chinese, English, and Greek characters and the Euro sign, enter the following:

```
myTextVar.text = "\u304B\uD55C\u6C49hello\u03BB\u20AC";
```

When the SWF file plays, the following characters appear in the text field:



か한汉helloλ€

For best results when creating a text field that contains multiple languages, use a font that includes all the glyphs your text needs.

See also

"Non-Unicode external files" on page 279

Using the XMLConnector component to connect to external XML files

Use the version 2 XMLConnector component to connect to an external XML document to bind to properties in the document. Its purpose is to read or write XML documents by using HTTP `GET` operations, `POST` operations, or both. It acts as a connector between other components and external XML documents. The XMLConnector communicates with components in your application by using either data-binding features in the Flash authoring environment or ActionScript code. For more information, see “XMLConnector component” in the *ActionScript 2.0 Components Language Reference*.

Chapter 14: Working with sound

You can use sound in Adobe® Flash® CS3 Professional in several different ways to make your work more interesting and involving. You can import sounds and edit them after they are imported. You can attach sounds to different kinds of objects and trigger them in different ways, depending on your desired effect.

Using sounds in Flash

About sounds and Flash

Adobe® Flash® CS3 Professional offers several ways to use sound. Make sounds that play continuously, independent of the Timeline, or use the Timeline to synchronize animation to a sound track. Add sounds to buttons to make them more interactive, and make sounds fade in and out for a more polished sound track.

There are two types of sounds in Flash: event sounds and stream sounds. An event sound must download completely before it begins playing, and it continues playing until explicitly stopped. Stream sounds begin playing as soon as enough data for the first few frames has been downloaded; stream sounds are synchronized to the Timeline for playing on a website.

If you're creating Flash content for mobile devices, Flash also lets you include device sounds in your published SWF file. Device sounds are encoded in the device's natively supported audio format, such as MIDI, MFi, or SMAF.

You can use shared libraries to link a sound to multiple documents. You can also use the ActionScript™ 2.0 `onSoundComplete` event to trigger an event based on the completion of a sound.

You can load sounds and control sound playback using prewritten behaviors or media components; the latter also provide a controller for stop, pause, rewind, and so on. You can also use ActionScript 2.0 or 3.0 to load sounds dynamically.

For more information, see `attachSound (Sound.attachSound method)` and `loadSound (Sound.loadSound method)` in *ActionScript 2.0 Language Reference* or `Sound class` in *ActionScript 3.0 Language and Components Reference*.

See also

"Using shared library assets" on page 218

Importing sounds

You place sound files into Flash by importing them into the library for the current document.

You can import the following sound file formats into Flash:

- WAV (Windows only)
- AIFF (Macintosh only)
- mp3 (Windows or Macintosh)

If you have QuickTime 4 or later installed on your system, you can import these additional sound file formats:

- AIFF (Windows or Macintosh)

- Sound Designer II (Macintosh only)
- Sound Only QuickTime Movies (Windows or Macintosh)
- Sun AU (Windows or Macintosh)
- System 7 Sounds (Macintosh only)
- WAV (Windows or Macintosh)

Flash stores sounds in the library along with bitmaps and symbols. You need only one copy of a sound file to use that sound multiple ways in your document.

If you want to share sounds among Flash documents, you can include the sounds in shared libraries.

Sounds can use large amounts of disk space and RAM. However, mp3 sound data is compressed and smaller than WAV or AIFF sound data. Generally, when using WAV or AIFF files, it's best to use 16-22 kHz mono sounds (stereo uses twice as much data as mono), but Flash can import either 8- or 16-bit sounds at sample rates of 11, 22, or 44 kHz. Sounds recorded in formats that are not multiples of 11 kHz (such as 8, 32, or 96 kHz) are resampled when imported into Flash. Flash can convert sounds to lower sample rates on export.

If you want to add effects to sounds in Flash, it's best to import 16-bit sounds. If you have limited RAM, keep your sound clips short or work with 8-bit sounds instead of 16-bit sounds.

See also

"Work with common libraries" on page 68

Import a sound

1 Select File > Import > Import To Library.

2 In the Import dialog box, locate and open the desired sound file.

***Note:** You can also drag a sound from a common library into the library for the current document.*

Add a sound to the Timeline

You can add a sound to a document using the library, or you can load a sound into a SWF file during runtime, using the `loadSound` method of the Sound object. For more information, see `loadSound (Sound.loadSound method)` in the *ActionScript 2.0 Language Reference* or `Sound Class` in the *ActionScript 3.0 Language and Components Reference*.

1 Import the sound into the library if it has not already been imported.

2 Select Insert > Timeline > Layer.

3 With the new sound layer selected, drag the sound from the Library panel onto the Stage. The sound is added to the current layer.

You can place multiple sounds on one layer or on layers containing other objects. However, it is recommended that each sound be placed on a separate layer. Each layer acts as a separate sound channel. The sounds on all layers are combined when you play the SWF file.

4 In the Timeline, select the first frame that contains the sound file.

5 Select Window > Properties > Properties, and click the arrow in the lower-right corner to expand the Property inspector.

6 In the Property inspector, select the sound file from the Sound pop-up menu.

7 Select an effect option from the Effects pop-up menu:

None Applies no effects to the sound file. Select this option to remove previously applied effects.

Left Channel/Right Channel Plays sound in the left or right channel only.

Fade Left To Right/Fade Right To Left Shifts the sound from one channel to the other.

Fade In Gradually increases the volume of a sound over its duration.

Fade Out Gradually decreases the volume of a sound over its duration.

Custom Lets you create custom in and out points of sound using the Edit Envelope.

8 Select a synchronization option from the Sync pop-up menu:

***Note:** If you are placing the sound on a frame other than Frame 1 in the main Timeline, select the Stop option.*

Event Synchronizes the sound to the occurrence of an event. An event sound, such as a sound that plays when a user clicks a button, plays when its starting keyframe first appears and plays in its entirety, independently of the Timeline, even if the SWF file stops playing. Event sounds are mixed when you play your published SWF file. If an event sound is playing and the sound is instantiated again (for example, by the user clicking the button again), the first instance of the sound continues to play and another instance begins to play simultaneously.

Start The same as Event, except that if the sound is already playing, no new instance of the sound plays.

Stop Silences the specified sound.

Stream Synchronizes the sound for playing on a website. Flash forces animation to keep pace with stream sounds. If Flash can't draw animation frames quickly enough, it skips frames. Unlike event sounds, stream sounds stop if the SWF file stops playing. Also, a stream sound can never play longer than the length of the frames it occupies. Stream sounds are mixed when you publish your SWF file.

An example of a stream sound is the voice of a character in an animation that plays in multiple frames.

***Note:** If you use an mp3 sound as a stream sound, you must recompress the sound for export. You can export the sound as an mp3 file, with the same compression settings that it had on import.*

9 Enter a value for Repeat to specify the number of times the sound should loop, or select Loop to repeat the sound continuously.

For continuous play, enter a number large enough to play the sound for an extended duration. For example, to loop a 15-second sound for 15 minutes, enter 60. Looping stream sounds is not recommended. If a stream sound is set to loop, frames are added to the file and the file size is increased by the number of times the sound is looped.

10 To test the sound, drag the playhead over the frames containing the sound or use commands in the Controller or the Control menu.

Synchronize a sound with animation

To synchronize a sound with animation, you start and stop the sound at keyframes.

1 Add a sound to a document.

2 To synchronize this sound with an event in the scene, select a beginning keyframe that corresponds to the keyframe of the event in the scene. You can select any of the synchronization options.

3 Create a keyframe in the sound layer's Timeline at the frame where you want the sound to end. A representation of the sound file appears in the Timeline.

4 Select Window > Properties > Properties, and click the arrow in the lower right-corner to expand the Property inspector.

5 In the Property inspector, select the same sound from the Sound pop-up menu.

6 Select Stop from the Sync pop-up menu.

When you play the SWF file, the sound stops playing when it reaches the ending keyframe.

7 To play back the sound, simply move the playhead.

Add a sound to a button

You can associate sounds with the different states of a button symbol. Because the sounds are stored with the symbol, they work for all instances of the symbol.

1 Select the button in the Library panel.

2 Select Edit from the Panel menu in the upper-right corner of the panel.

3 In the button's Timeline, add a layer for sound (Insert > Timeline > Layer).

4 In the sound layer, create a regular or blank keyframe to correspond with the button state to which you want to add a sound (Insert > Timeline > Keyframe or Insert > Timeline > Blank Keyframe).

For example, to add a sound that plays when you click the button, create a keyframe in the frame labeled Down.

5 Click the keyframe you created.

6 Select Window > Properties > Properties.

7 In the Property inspector, select a sound file from the Sound pop-up menu.

8 Select Event from the Sync pop-up menu.

To associate a different sound with each of the button's keyframes, create a blank keyframe and add another sound file for each keyframe. You can also use the same sound file and apply a different sound effect for each button keyframe.

Edit a sound

You can define the starting point of a sound or to control the volume of the sound as it plays. You can also change the point at which a sound starts and stops playing. This is useful for making sound files smaller by removing unused sections.

1 Add a sound to a frame, or select a frame that already contains a sound.

2 Select Window > Properties > Properties.

3 Click the Edit button on the right side of the Property inspector.

4 Do any of the following:

- To change the start and end points of a sound, drag the Time In and Time Out controls in the Edit Envelope.
- To change the sound envelope, drag the envelope handles to change levels at different points in the sound. Envelope lines show the volume of the sound as it plays. To create additional envelope handles (up to eight total), click the envelope lines. To remove an envelope handle, drag it out of the window.
- To display more or less of the sound in the window, click the Zoom In or Out buttons.
- To switch the time units between seconds and frames, click the Seconds and Frames buttons.

5 To hear the edited sound, click the Play button.

Using sounds in Flash Lite

Adobe® Flash® Lite supports two types of sound: standard Flash sounds, like those used in Flash desktop applications, and device sounds. Flash Lite 1.0 supports device sounds only; Flash Lite 1.1 and 2.x support both standard sounds and device sounds.

Device sounds are stored in the published SWF file in their native audio format (such as MIDI or MFi); during playback, Flash Lite passes the sound data to the device, which decodes and plays the sound. Because you can't import most device audio formats into Flash, you instead import a *proxy* sound in a supported format (such as mp3 or AIFF) that is replaced with an external device sound that you specify.

You can use device sounds only as event sounds—you can't synchronize device sounds with the Timeline, as you can with standard sounds.

Flash Lite 1.0 and Flash Lite 1.1 do not support the following features available in the desktop version of Flash® Player:

- The ActionScript Sound object
- Loading of external mp3 files
- The Speech Audio Compression option

For more information, see "Working with Sound, Video, and Images" in *Developing Flash Lite 2.x Applications* or "Working with Sound" in *Developing Flash Lite 1.x Applications*.

Exporting Sounds

Compressing sounds for export

You can select compression options for individual event sounds and export the sounds with those settings. You can also select compression options for individual stream sounds. However, all stream sounds in a document are exported as a single stream file, using the highest setting of all those applied to individual stream sounds. This includes stream sounds in video objects.

If you select global compression settings for event sounds or stream sounds in the Publish Settings dialog box, these settings are applied to individual event sounds or all stream sounds if you do not select compression settings for the sounds in the Sound Properties dialog box.

You can also override export settings specified in the Sound Properties dialog box by selecting Override Sound Settings in the Publish Settings dialog box. This option is useful if you want to create a larger high-fidelity audio file for local use and a smaller low-fidelity version for the web.

The sampling rate and degree of compression make a significant difference in the quality and size of sounds in exported SWF files. The more you compress a sound and the lower the sampling rate, the smaller the size and the lower the quality. You should experiment to find the optimal balance between sound quality and file size.

When working with imported mp3 files, you can export the files in mp3 format using the same settings that the files had when imported.

Note: *In Windows, you can also export all the sounds from a document as a WAV file using File > Export > Export Movie.*

See also

“Publishing overview” on page 418

“Set publish options for the Flash SWF file format” on page 420

“Exporting Flash content, images, and video” on page 447

Compress a sound for export

1 Do one of the following:

- Double-click the sound’s icon in the Library panel.
- Right-click (Windows) or Control-click (Macintosh) a sound file in the Library panel and select Properties from the context menu.
- Select a sound in the Library panel and select Properties from the Panel menu in the upper right corner of the panel.
- Select a sound in the Library panel and click the Properties button at the bottom of the Library panel.

2 If the sound file has been edited externally, click Update.

3 For Compression, select Default, ADPCM, mp3, Raw, or Speech.

The Default compression option uses the global compression settings in the Publish Settings dialog box when you export your SWF file. If you select Default, no additional export settings are available.

4 Set export settings.

5 Click Test to play the sound once. Click Stop if you want to stop testing the sound before it finishes playing.

6 Adjust export settings if necessary until the desired sound quality is achieved, and then click OK.

ADPCM and Raw compression options

ADPCM compression sets compression for 8- or 16-bit sound data. Use the ADPCM setting when you export short event sounds such as button clicks.

Raw compression exports sounds with no sound compression.

Preprocessing Converts mixed stereo sounds to monaural (mono) when you select Convert Stereo To Mono (mono sounds are unaffected by this option).

Sample Rate Controls sound fidelity and file size. Lower rates decrease file size but can also degrade sound quality. Rate options are as follows:

- **5 kHz** Barely acceptable for speech.
- **11 kHz** The lowest recommended quality for a short segment of music and one-quarter the standard CD rate.
- **22 kHz** A popular choice for web playback and half the standard CD rate.
- **44 kHz** The standard CD audio rate.

Note: *Flash cannot increase the kHz rate of an imported sound above the rate at which it was imported.*

mp3 compression options

MP3 Compression lets you export sounds with mp3 compression. Use mp3 when you are exporting longer stream sounds such as music sound tracks.

If you are exporting a file that was imported in mp3 format, you can export the file using the same settings the file had when it was imported.

Use Imported mp3 Quality Default setting. Deselect to select other mp3 compression settings. Select to export an imported mp3 file with the same settings the file had when it was imported.

Bit Rate Determines the bits per second in the exported sound file. Flash supports 8 through 160 Kbps CBR (constant bit rate). When you export music, set the bit rate to 16 Kbps or higher for best results.

Preprocessing Converts mixed stereo sounds to monaural (mono sounds are unaffected by this option).

***Note:** The Preprocessing option is available only if you select a bit rate of 20 Kbps or higher.*

Quality Determines the compression speed and sound quality:

- **Fast** Yields faster compression but lower sound quality.
- **Medium** Yields somewhat slower compression but higher sound quality.
- **Best** Yields the slowest compression and the highest sound quality.

Speech compression option

Speech compression exports sounds using a compression that is adapted to speech.

***Note:** Flash Lite 1.0 and Flash Lite 1.1 do not support the Speech compression option. For content targeting those player versions, use mp3, ADPCM, or Raw compression.*

Sample rate Controls sound fidelity and file size. A lower rate decreases file size but can also degrade sound quality. Select from the following options:

- **5 kHz** Acceptable for speech.
- **11 kHz** Recommended for speech.
- **22 kHz** Acceptable for most types of music on the web.
- **44 kHz** The standard CD audio rate. However, because compression is applied, the sound is not CD quality in the SWF file.

Guidelines for exporting sound in Flash documents

Besides sampling rate and compression, there are several ways to use sound efficiently in a document and keep file size small:

- Set the in and out points to prevent silent areas from being stored in the Flash file and to reduce the size of the sound.
- Get more out of the same sounds by applying different effects for sounds (such as volume envelopes, looping, and in/out points) at different keyframes. You can get a number of sound effects using only one sound file.
- Loop short sounds for background music.
- Do not set streaming sound to loop.
- When exporting audio in embedded video clips, remember that the audio is exported using the global streaming settings selected in the Publish Settings dialog box.
- Use stream synchronization to keep the animation synchronized to your sound track when you preview your animation in the editor. If your computer is not fast enough to draw the animation frames so that they keep up with your sound track, Flash skips frames.

- When exporting QuickTime movies, use as many sounds and channels as you want without worrying about file size. The sounds are combined into a single sound track when you export as a QuickTime file. The number of sounds you use has no effect on the final file size.

Sound and ActionScript

Control sounds using behaviors

Using sound behaviors, prewritten ActionScript 2.0, you can add sounds to your document and control sound playback. Adding a sound using these behaviors creates an instance of the sound, which is then used to control the sound.

***Note:** ActionScript 3.0 and Flash Lite 1.x and Flash Lite 2.x do not support behaviors.*

Load a sound to a file using a behavior

1 Select the object, such as a button, that you want to use to trigger the behavior.

2 In the Behaviors panel (Window > Behaviors), click the Add (+) button and select Sound > Load Sound from Library or Sound > Load Streaming mp3 File.

3 In the Load Sound dialog box, enter the linkage identifier for a sound from the Library, or the sound location for a streaming mp3 file. Next, enter a name for this instance of the sound, and click OK.

4 In the Behaviors panel under event, click On Release (the default event), and select a mouse event from the menu. If you want to use the `OnRelease` event, do not change the option.

Play or stop sounds using a behavior

1 Select the object, such as a button, that you want to use to trigger the behavior.

2 In the Behaviors panel (Window > Behaviors), click the Add (+) button.

3 Select Sound > Play Sound, Sound > Stop Sound, or Sound > Stop All Sounds.

4 In the dialog box that appears, do one of the following:

- Enter the linkage identifier and the instance name of the sound you want to play or stop, and click OK.
- Click OK to verify that you want to stop all sounds.

5 In the Behaviors panel under Event, click On Release (the default event) and select a mouse event from the menu. If you want to use the `OnRelease` event, do not change the option.

Control sounds with the ActionScript 2.0 Sound object

Use the Sound object in ActionScript 2.0 to add sounds to a document and to control sound objects in a document, including adjusting the volume or the right and left balance while a sound plays. For more information, see Creating sound controls in *Learning ActionScript 2.0 in Adobe Flash*.

1 Select the sound in the Library panel.

2 Select Linkage from the Panel menu in the upper-right corner of the panel, or right-click (Windows) or Control-click (Macintosh) the sound name in the Library panel and select Linkage from the context menu.

3 Under Linkage in the Linkage Properties dialog box, select Export for ActionScript.

4 Enter an identifier string in the box, and click OK.

Accessing ID3 properties in mp3 files with Flash Player

Macromedia Flash Player 7 from Adobe and later supports ID3 v2.4 and v2.4 tags. With this version, when you load an mp3 sound using the ActionScript 2.0 `attachSound()` or `loadSound()` method, the ID3 tag properties are available at the beginning of the sound data stream. The onID3 event executes when the ID3 data is initialized.

Flash Player 6 (6.0.40.0) and later supports mp3 files with ID3 v1.0 and v1.1 tags. With ID3 v1.0 and v1.1 tags, the properties are available at the end of the data stream. If a sound does not contain an ID3v1 tag, the ID3 properties are undefined. Users must have Flash Player 6 (6.0.40.0) or later for the ID3 properties to function.

For more information on using the ID3 properties, see `id3` (`Sound.id3` property) in the *ActionScript 2.0 Language Reference*.

About the ActionScript 2.0 onSoundComplete event

The `onSoundComplete` event of the ActionScript 2.0 Sound object lets you trigger an event in a Flash application based on completing an attached sound file. The Sound object is a built-in object that lets you control sounds in a Flash application. For more information, see Sound in the *ActionScript 2.0 Language Reference*. The `onSoundComplete` event of a Sound object is invoked automatically when the attached sound file finishes playing. If the sound is looped a specified number of times, the event is triggered when the sound finishes looping.

The Sound object has two properties that you can use with the `onSoundComplete` event. The `duration` property is a read-only property representing the duration, in milliseconds, of the sound sample attached to the sound object. The `position` property is a read-only property representing the number of milliseconds the sound has been playing in each loop.

The `onSoundComplete` event lets you manipulate sounds in a many ways, such as the following:

- Creating a dynamic playlist or sequencer
- Creating a multimedia presentation that checks for narration completion before advancing to the next frame or scene
- Building a game that synchronizes sounds to particular events or scenes and transitions smoothly between different sounds
- Timing an image change to a sound—for example, changing an image when a sound is halfway through at playback time

Chapter 15: Working with video

Adobe® Flash® CS3 Professional is a powerful tool for incorporating video footage into web-based presentations. Flash Video offers technological and creative benefits that let you fuse video together with data, graphics, sound, and interactive control. Flash Video lets you easily put video on a web page in a format that almost anyone can view.

Creating and publishing Flash Video

About video features in Flash

How you choose to deploy your video determines how you create your video content, and how you integrate it for use with Flash. You can incorporate video into Flash in the following ways:

Streaming video content Lets you host video files using Flash Media Server, a server solution optimized to deliver streaming, real-time media. Import video clips stored locally into your Flash documents, and later upload them to the server, to more easily assemble and develop Flash content. To control video playback and provide intuitive controls for users to interact with the video, use the new FLVPlayback component or ActionScript™. Host your own Flash Media Server, or use a hosted Flash Video Streaming Service (FVSS). Adobe has partnered with several content delivery network (CDN) providers to offer hosted services for delivering on-demand Flash Video across high-performance, reliable networks. Built with Flash Media Server and integrated directly into the delivery, tracking, and reporting infrastructure of the CDN network, FVSS provides the most effective way to deliver Flash Video to the largest possible audience without the hassle of setting up and maintaining your own streaming server hardware and network.

Progressively downloading video from a web server If you don't have access to Flash Media Server or FVSS, you can still download video from an external source when you use progressive downloading. Progressively downloading a video clip from a web server doesn't provide the same real-time performance that Flash Media Server does; however, you can use relatively large video clips, and keep the size of your published SWF files to a minimum. To control video playback and provide intuitive controls for users to interact with the video, use the new FLVPlayback component or ActionScript.

Importing embedded video Import video clips into Flash as embedded files. An embedded video file becomes part of the Flash document. For this reason, you can only import short-duration video clips.

Importing video in QuickTime format Import video clips in QuickTime format as linked files. Flash documents that contain linked QuickTime video must be published in QuickTime format. A linked video file does not become part of the Flash document. Instead, the document maintains a pointer to the linked file.

Importing FLV files in the Library Import video clips in Flash Video (FLV) format from Adobe® directly into Flash. When you import FLV files, you use the encoding options already applied to the files. You do not need to select encoding options during import.

You can control the playback of video files in the following ways:

Using the FLVPlayback component Lets you quickly add a full-featured FLV or mp3 playback control to your Flash movie and provides support for both progressive downloading and streaming FLV files. FLVPlayback lets you easily create intuitive video controls for users to control video playback, and apply premade skins, or apply your own custom skins to the video interface.

Controlling external video playback using ActionScript Play back external FLV files in a Flash document at runtime using the `NetConnection` and `NetStream` ActionScript objects.

You can use video behaviors (prewritten ActionScript scripts) to control video playback.

Controlling video playback in the Timeline To control video playback, write custom ActionScript.

For video tutorials about working with video in Flash, see the following:

- Using video in Flash: www.adobe.com/go/vid0136
- Using Flash Video Encoder: www.adobe.com/go/vid0138
- Creating content for Adobe After Effects: www.adobe.com/go/vid0139

For a text tutorial about using video in Flash, see Building a Video Player on the Flash Tutorials page at www.adobe.com/go/learn_fl_tutorials.

See also

“Import linked QuickTime video files” on page 307

“Importing and modifying Flash Video files” on page 301

“About the FLVPlayback component” on page 322

“Playing back external FLV files dynamically” on page 320

“Control video playback using behaviors” on page 322

Importing and modifying Flash Video files

Import Flash Video files into the library

To import files in the FLV format, use the Import or Import To Library commands or the Import button in the Video Properties dialog box.

To create your own video player, which dynamically loads FLV files from an external source, place your video inside a movie clip symbol. When you load FLV files dynamically, adjust the dimensions of the movie clip to match the actual dimension of the FLV and scale the video by scaling the movie clip.

***Note:** A best practice is to place video inside a movie clip instance, which gives you the most control over the content. The video’s Timeline plays independently from the main Timeline. You do not have to extend your main Timeline by many frames to accommodate the video, which can make working with your FLA file difficult.*

❖ To import an FLV file into the Library, do one of the following:

- Select File > Import > Import To Library.
- Select any existing video clip in the Library panel, and select Properties from the Library Panel menu. Click Import. Locate the file to import, and click Open.

Controlling video playback using the Timeline

To control playback of an embedded or linked video file, control the Timeline that contains the video. For example, to pause a video playing on the main Timeline, you would call a `stop()` action that targets that Timeline. Similarly, you can control a video object in a movie clip symbol by controlling the playback of that symbol’s Timeline.

You can apply the following actions to imported video objects in movie clips: `goTo`, `play`, `stop`, `toggleHighQuality`, `stopAllSounds`, `getURL`, `FScommand`, `loadMovie`, `unloadMovie`, `ifFrameLoaded`, and `onMouseEvent`. To apply actions to a Video object, first convert the Video object to a movie clip.

To show a live video stream from a camera, use ActionScript. First, to place a Video object on the Stage, use the New Video Object in the Library panel. To attach the video stream to the Video object, use `Video.attachVideo`.

See also Video and attachVideo (Video.attachVideo method) in the *ActionScript 2.0 Language Reference*, and fl.video in the ActionScript 3.0 Language Reference.

See also

"Playing back external FLV files dynamically" on page 320

About progressively downloading video

Progressive downloading lets you use ActionScript to load external FLV files into a SWF file, and play them back at runtime. To initiate playback of the FLV file, and to control the Play, Pause, and Seek behaviors, as well as the buffer time and size for a given video file, use the `netConnection` and `netStream` objects.

Because the video content is kept external to the other Flash content and the video playback controls, it's relatively easy to update video content without republishing the SWF file.

Progressive downloading provides the following advantages over embedded video:

- During authoring, publish only the SWF interface to preview or test part or all of your Flash content. This results in faster preview times and quicker turnaround on iterative experimentation.
- During delivery, video begins playing as soon as the first segment is downloaded and cached to the local computer's disk drive.
- At runtime, video files are loaded from the computer's disk drive into the SWF file, with no limitation on file size or duration. No audio synchronization issues or memory restrictions exist.
- The frame rate of the video file can be different from the frame rate of the SWF file, allowing for greater flexibility in authoring your Flash content.

Streaming video using Flash Media Server

In streaming, each Flash client opens a persistent connection to the Flash Media Server, and a controlled relationship exists between the video being delivered and the client interaction. Flash Media Server uses bandwidth detection to deliver video or audio content based on the user's available bandwidth. This lets you provide different content for users based on their ability to easily access and download content. For example, if a user with a dial-up modem accesses your video content, you can deliver an appropriately encoded file that doesn't require too much bandwidth.

Flash Media Server also provides you with quality of service metrics, detailed tracking and reporting statistics, and a range of interactive features designed to enhance the video experience. As with progressive downloading, the video content (FLV file) is kept external to the other Flash content and the video playback controls. This lets you easily add or change content without having to republish the SWF file.

Streaming video with Flash Media Server or FVSS provides the following advantages over embedded and progressively downloaded video:

- Video playback starts sooner than it does using other methods of incorporating video.
- Streaming uses less of the client's memory and disk space, because the clients don't need to download the entire file.

- Network resources are used more efficiently, because only the parts of the video that are viewed are sent to the client.
- Delivery of media is more secure, because media is not saved to the client's cache when streamed.
- Streaming video provides better tracking, reporting, and logging ability.
- Streaming lets you deliver live video and audio presentations, or capture video from a web cam or digital video camera.
- Flash Media Server enables multiway and multiuser streaming for video chat, video messaging, and video conferencing applications.
- By using server-side scripting to control video and audio streams, you can create server-side play lists, synchronized streams, and more intelligent delivery options based on the client's connection speed.

To learn more about Flash Media Server, see: www.adobe.com/go/flash_media_server.

To learn more about FVSS, see: www.adobe.com/go/fvss.

Embedding video in a SWF file

Embedded video lets you embed a video file within a SWF file. When you import video in this way, the video is placed in the Timeline where you can see the individual video frames represented in the Timeline frames. An embedded video file becomes part of the Flash document.

When you create a SWF file with embedded video, the frame rate of the video clip and the SWF file must be the same. If you use different frame rates for the SWF file and the embedded video clip, playback is inconsistent. To use variable frame rates, import the video using either progressive download or Flash Media Server. When you import video files using either of these methods, the FLV files are self-contained and run at a frame rate separate from that of all other timeline frame rates included in the Flash SWF file.

You can import video clips into Flash as embedded files in QuickTime video (MOV), Audio Video Interleaved file (AVI), Motion Picture Experts Group file (MPEG), or other formats, depending on your system.

Embedded video works best for smaller video clips, with a playback time of less than 10 seconds. If you are using video clips with longer playback times, consider using progressively downloaded video, or streaming video using Flash Media Server.

The limitations of embedded video include:

- You might encounter problems if the resulting SWF files become excessively large. Flash Player reserves a lot of memory when downloading and attempting to play large SWF files with embedded video, which can cause Flash Player to fail.
- Longer video files (over 10 seconds long) often have synchronization issues between the video and audio portions of a video clip. Over time, the audio track begins playing out of sequence with the video, causing a less than desirable viewing experience.
- To play a video embedded in a SWF file, the entire video file must be downloaded before the video starts to play. If you embed an excessively large video file, it might take a long time for the SWF file to download in its entirety and for playback to start.

See also

"Supported file formats for video" on page 309

About Linked QuickTime video

Using Flash, you can create QuickTime movies (MOV files) that can be played back by users who have the QuickTime plug-in installed on their computers. This is often done when people use Flash to create title sequences or animation for use as video content. The published QuickTime file can be distributed as a DVD, or incorporated into other applications such as Adobe® Director® or Adobe® Premiere®.

When you use Flash to create a QuickTime video clip, you can link to a QuickTime video from the Flash file rather than embed the video. A linked QuickTime video imported into Flash does not become part of the file. Instead, a pointer is maintained to the source file.

If you are creating a QuickTime video using Flash, set your publish setting to Flash 3, 4, or 5. You cannot display a linked QuickTime video in SWF format. The QuickTime file contains a Flash track, but the linked video clip remains in QuickTime format.

***Note:** Importing a video clip as linked QuickTime means that the resulting content can only be published as a QuickTime MOV file. You cannot publish content using a linked QuickTime video as a SWF file.*

Change the properties of a video clip

To modify embedded and linked video clips, use the Property inspector and the Video Properties dialog box.

In the Property inspector, you can change properties for an instance of an embedded or linked video clip on the Stage, assign the instance an instance name, and change its width, height, and position on the Stage. You can also *swap* an instance of a video clip—assign a different symbol to an instance of a video clip. Assigning a different symbol to an instance displays a different instance on the Stage but leaves all the other instance properties (such as dimensions and registration point) intact.

In the Video Properties dialog box, you can do the following:

- View information about an imported video clip, including its name, path, creation date, pixel dimensions, length, and file size
- Change the video clip name
- Update the video clip if you modify it in an external editor
- Import an FLV video to replace the selected clip
- Export a video clip as an FLV file

For lessons on working with video, see the Adobe Flash Support Center at www.adobe.com/go/flash_video.

Change video instance properties in the Property inspector

1 Select an instance of an embedded or linked video clip on the Stage.

2 Select Window > Properties > Properties, and do any of the following:

- Enter an instance name in the Name text field on the left side of the Property inspector.
- Enter values for W and H to change the dimensions of the video instance.
- Enter values for X and Y to change the position of the upper-left corner of the instance on the Stage.
- Click Swap. Select a video clip to replace the clip currently assigned to the instance.

***Note:** You can swap an embedded video clip only with another embedded video clip, and you can swap a linked video clip only with another linked video clip.*

View video clip properties in the Video Properties dialog box

1 Select a video clip in the Library panel.

2 Select Properties from the Library Panel menu, or click the Properties button located at the bottom of the the Library panel. The Video Properties dialog box is displayed.

Assign a new name to, update, or replace a video clip with an FLV clip

1 Select the video clip in the Library panel and select Properties from the Library Panel menu.

2 Do one of the following:

- To assign a new name, enter the name in the Name text field.
- To update a video clip, navigate to the updated video file and click Open.
- To replace a video clip with an FLV clip, click Import, navigate to the FLV file to replace the current clip, and click Open.

Import video with Flash Media Server or FVSS

Import a video file that is already deployed to a web server or a Flash Media Server or Flash Video Streaming Service (FVSS), or select a video file that is stored locally on your computer, and upload the video file to the server after importing it to your FLA file.

1 To import the video clip into the current Flash document, select File > Import > Import Video.

2 Select the video clip to import. Select either a video clip stored on your local computer, or enter the URL of a video already uploaded to a web server, your own Flash Media Server, or an FVSS and do one of the following:

- To import video for progressive download, select Progressive Download from a standard web server.
- To import video for streaming with Flash Media Server or FVSS, select Stream From Flash Video Streaming Service (FVSS) or Stream From Flash Media Server (FMS).

3 (Optional) If the video you are deploying is not in FLV format, use the Encoding panel to select an encoding profile, and copy, trim, and split the video clip.

***Note:** This step applies only if you are uploading the video from your local computer. Video clips that are already deployed to a server must be previously encoded in the FLV format.*

4 Select a skin for your video clip. You can choose to:

- Not use a skin with the video.
- Select one of the predefined skins.
- Select a custom skin of your own design by entering the URL of the skin on the server.

The Video Import wizard encodes your source video clip in the FLV format (if it isn't already in FLV format) and creates a video component on the Stage that you can use to test video playback locally.

5 Upload the following assets to the web server hosting your video:

- The FLV encoded video clip (which is located in the same folder as the source video clip you selected with a .flv extension)

***Note:** If the video clip is in FLV format, Flash uses a relative path to point to the FLV file (relative to the SWF file), letting you use the same directory structure locally that you use on the server. If the video was previously deployed to your FMS or FVSS hosting your video, you can skip this step.*

- The video skin (if you chose to use a skin)

To use a predefined skin, Flash copies the skin into the same folder as the FLA file.

- The video component

To edit the component’s URL field to that of the web server that you are uploading the video to, use the Component inspector.

See also

“Select a video encoding profile” on page 313

“Specify the contentPath parameter” on page 323

“About the FLVPlayback component” on page 322

Embed video in a SWF file

Import video clips as embedded files in several file formats, depending on your system. Preview frames of an imported video by dragging the playhead along the Timeline. However, the sound does not play back. To preview the video with sound, use the Test Movie command.

When you import a video as an embedded file, you can edit the video before importing it. You can also apply customized compression settings, including bandwidth or video quality settings.

***Note:** After a video clip is imported, it cannot be edited.*

See also

“Supported file formats for video” on page 309

“Test document download performance” on page 492

“Types of symbols” on page 208

“Select a video encoding profile” on page 313

Embed video within the SWF file

1 To import the video clip into the current Flash document, select File > Import > Import Video.

2 Select the video clip on your local computer to import.

3 Select Embed Video In SWF and Play In Timeline.

4 Choose the symbol type with which to embed the video in the SWF file.

Embed in the Timeline If you’re using the video clip for linear playback in the Timeline, importing the video into the Timeline is the most appropriate method.

Embed as a movie clip A best practice is to place video inside a movie clip instance, because you have the most control over the content. The video’s Timeline plays independently from the main Timeline. You do not have to extend your main Timeline by many frames to accommodate the video, which can make working with your FLA file difficult.

Embed as a graphic symbol When you embed a video clip as a graphic symbol, you cannot interact with the video using ActionScript (typically you use graphic symbols for static images and to create reusable pieces of animation that are tied to the main Timeline).

5 Import the video clip directly onto the Stage (and the Timeline) or as a library item.

By default, Flash places the video you import on the Stage. To import into the library only, deselect Place Instance on Stage.

If you're creating a simple video presentation with linear narration and little to no interaction, accept the default setting and import the video to the Stage. To create a more dynamic presentation, work with multiple video clips, or add dynamic transitions or other elements using ActionScript, import the video into the library. After a video clip is in the library, customize it by converting it into a MovieClip object that you can more easily control with ActionScript.

By default, Flash expands the Timeline to accommodate the playback length of the video clip you are embedding.

6 (Optional) To edit your video clip before embedding it in the Timeline, select Edit Video First.

7 (Optional) If the video clip is not yet encoded in the FLV format, select a Flash Video encoding profile.

8 Click Finish.

The Video Import wizard encodes your video into the FLV format, and embeds the video into the SWF file. The video appears either on the Stage or in the library depending on the embedding options you chose.

9 In the Property inspector (Window > Properties), give the video clip an instance name, and make any modifications to the video clip's properties.

Update an embedded video clip after editing it in an external editor

1 Select the video clip in the Library panel.

2 Select Properties and click Update.

The embedded video clip is updated with the edited file. The compression settings you selected when you first imported the video are reapplied to the updated clip.

Import linked QuickTime video files

A linked QuickTime video imported into Flash does not become part of the Flash file. Instead, Flash maintains a pointer to the source file.

If you link to a QuickTime video, publish the SWF file as a QuickTime video. You cannot display a linked QuickTime clip in SWF format. The QuickTime file contains a Flash track, but the linked video clip remains in QuickTime format.

You can scale, rotate, and animate a linked QuickTime video in Flash. However, you cannot tween linked QuickTime video content in Flash.

Note: *The QuickTime Player does not support Flash Player files later than version 5.*

See also

"About QuickTime" on page 452

"Specify publish settings for QuickTime videos" on page 430

Import a QuickTime video as a linked file

1 Do one of the following:

- To link the video clip directly to the current Flash document, select File > Import > Import To Stage.
- To link the video clip to the library for the current Flash document, select File > Import > Import To Library.

2 Select Link To External Video File, and click Next.

3 If you imported the video clip directly to the Stage in step 1, a warning appears if the imported clip contains more frames than the span in which you are placing it in the current Flash document. Do one of the following:

- To extend the span the required number of frames, click Yes.
- To keep the span at its current size, click No. Frames in the imported clip that exceed the frames in the span do not appear unless you subsequently add frames to the span.

4 Select Control > Play. Preview a linked QuickTime video before you publish your SWF file. When you import a linked QuickTime video, Flash adds the required number of frames to preview the QuickTime video, the same as it does for an embedded video.

Note: *You cannot use the Test Movie command to preview linked QuickTime video content.*

Change the directory path of a linked QuickTime video

1 Select Window > Library, and select the desired linked QuickTime video.

2 In the Panel menu in the Library panel, select Properties.

3 Click Set Path.

4 Navigate to the file for the linked video clip, select it, click Open, and click OK.

Edit video clips in the Video Import wizard

1 Import the video clip.

2 To open the Editing pane of the Video Import wizard, select Edit the Video First, and click Next.

3 To browse frames in the video, do one of the following:

- Drag the playhead along the scrubber bar.
- To move forward, click the Play button; to stop at the desired frame, click the Pause button.
- To move forward or backward one frame at a time, click the Backward and Forward buttons in the Controller.

4 To set the in and out points (beginning and ending frames), do one of the following:

- Drag the in and out points (the triangles below the scrubber bar).
- To set the beginning or ending frame at the current location of the playhead, click the In or Out button below the scrubber bar.

5 To play the video, do one of the following:

- To play the video from the current playhead position, click the Play button in the button controls.
- To play the video with the current in and out points, click Preview.

Note: *To stop video playback, click the Stop button.*

6 To create a clip with the current in and out points, click Create Clip.

To create additional clips from the same file, select in and out points for the clips as described in step 4, and click Create Clip again.

7 To rename a clip, select it in the scroll pane and enter the new name.

8 To re-edit a clip, select it in the scroll pane. Select new in and out points as described in step 4, and click Update Clip.

9 To delete a clip from the scroll pane, select the clip and click the Delete (-) button.

10 When the editing process is complete, click Next.

About digital video and Flash

Supported file formats for video

If QuickTime 7 for Macintosh, QuickTime 6.5 for Windows, or DirectX 9 or later (Windows only) is installed on your system, you can import video clips in several file formats, including MOV, AVI, and MPG/MPEG. You can import linked video clips in MOV format.

Flash documents with embedded video can be published as SWF files. Flash documents with linked video must be published in QuickTime format.

The following video file formats are supported for importing embedded video if QuickTime 7 is installed on your Macintosh computer:

File type	Extension
Audio Video Interleaved	.avi
Digital video	.dv
Motion Picture Experts Group	.mpg, .mpeg
QuickTime Video	.mov

The following video file formats are supported for importing embedded video if DirectX 9 or later is installed (Windows only):

File type	Extension
Audio Video Interleaved	.avi
Motion Picture Experts Group	.mpg, .mpeg
Windows Media file	.wmv, .asf

By default, Flash imports and exports video using the On2 VP 6 *codec*. A codec is a compression-decompression algorithm that controls how multimedia files are compressed during encoding, and decompressed during playback.

If you attempt to import a file format that is not supported on your system, a warning message appears that the operation cannot be completed. In some cases, Flash might import the video but not the audio in a file. For example, audio is not supported in MPG and MPEG files imported with QuickTime 7 on Mac OS. In such cases, a warning indicates that the audio portion of the file cannot be imported. You can import the video without sound.

See also

“About Linked QuickTime video” on page 304

“Comparing the On2 VP6 and Sorenson Spark video codecs” on page 310

Audio support for MPEG video

Because MPEG encodes both the video and audio portions of a file into a single track, encoding MPEG files as FLV files might result in the audio portion being removed. This occurs primarily when encoding video files into FLV format on Mac OS. On Mac OS, MPEG video is imported using QuickTime. QuickTime does not support extracting audio content from MPEG files (although QuickTime correctly plays back an MPEG file with audio programming).

If you are encoding MPEG video with audio on a Macintosh computer, it is recommended that you first convert the MPEG video clip into another format that encodes audio and video as separate tracks within the file. You can then encode the other format as an FLV file, and preserve the audio content.

An alternative is to use a computer with the Windows operating system. Windows imports MPEG video using DirectShow, which supports extracting both the video and audio tracks of an MPEG file. This lets you convert MPEG video to the FLV format without removing the audio portion of the video clip.

Note: *Imported audio is published or exported as streamed audio, using the global audio streaming settings selected in the Publish Settings dialog box.*

See also

"Set publish options for the Flash SWF file format" on page 420

Comparing the On2 VP6 and Sorenson Spark video codecs

The On2 VP6 codec is the default video codec to use when encoding Flash content that uses video and FLV content to use with Flash Player 8 and 9. The On2 VP6 codec provides:

- Higher quality video when compared to the Sorenson Spark codec encoded at the same data rate
- Support for the use of an alpha channel to composite video

To support better quality video at the same data rate, the On2 VP6 codec is noticeably slower to encode and requires more processor power on the client computer to decode and play back. For this reason, carefully consider the lowest common denominator of computer you intend your viewing audience to use when accessing your Flash Video content.

If you anticipate a large user base that uses older computers, consider encoding your FLV files using the Sorenson Spark codec.

If your Flash content dynamically loads Flash video (using either progressive download or Flash Media Server), you can use On2 VP6 video without having to republish your SWF file for Flash Player 8, as long as users use Flash Player 8 or later to view your content. By streaming or downloading On2 VP6 video into Flash SWF versions 6 or 7, and playing the content using Flash Player 8 or later, you avoid having to recreate your SWF files for use with Flash Player 8 and later versions.

Important: *Only Flash Player 8 and 9 supports both publish and playback of On2 VP6 video.*

Codec	Content (SWF) version (publish version)	Flash Player version (version required for playback)
Sorenson Spark	6	6, 7, 8
	7	7, 8, 9
On2 VP6	6, 7, 8	8, 9
	9	9

Tips for creating Flash video

Follow these guidelines to deliver the best possible Flash video:

Whenever possible, always encode a file from its uncompressed form

If you convert a precompressed digital video format into the FLV format, the previous encoder can introduce video noise. The first compressor already performed its encoding algorithm on the video and reduced its quality, frame size, and rate. It might have also introduced some of its own digital artifacts or noise. This additional noise affects the FLV encoding process and might require a higher data rate to play back a good-quality file.

Strive for simplicity Avoid elaborate transitions—they don't compress well and can make your final compressed video look chunky during the change. Hard cuts (as opposed to cross-dissolves) are usually best. Video sequences that show an object zooming from behind the first track, doing a page turn, or wrapping around a ball and then flying off the screen can be eye-catching, but they usually don't compress well and should be used sparingly.

Know your audience data rate

When you deliver video over the Internet, produce files at lower data rates. Users with fast Internet connections can view the files with little or no wait, but dial-up users must wait for files to download. Make the clips short to keep the download times within acceptable limits for dial-up users.

Select the proper frame rate

Frame rate indicates frames per second (fps). If you have a higher data rate clip, a lower frame rate can improve playback on lower-end computers. For example, if you are compressing a clip with little motion, cutting the frame rate in half probably saves you only 20 percent of the data rate. However, if you are compressing high-motion video, reducing the frame rate has a much greater effect on the data rate.

Because video looks much better at native frame rates, Adobe® recommends leaving the frame rate high if your delivery channels and playback platforms allow it. However, if you need to reduce the frame rate, the best results come from dividing the frame rate by whole numbers.

***Note:** To embed video clips in the SWF file, the frame rate of the video clip must be the same that the SWF file uses. To encode video using the same frame rate as the FLA file, use the Advanced Video Encoding settings in the Video Import wizard.*

Select a frame size that fits your data rate

At a given data rate (connection speed), increasing the frame size results in decreased video quality. When you select the frame size for your document, consider frame rate, source material, and personal preferences. Use the following list of common frame sizes (in pixels) as a guide. Experiment to find the best setting for your project.

- Modem: 160 x 120
- Dual ISDN: 192 x 144
- T1/DSL/cable: 320 x 240

Know progressive download times

Know how long it will take to download your video. While your video clip downloads, you might want to have other content that appears and disguises the download. For short clips, use the following formula: Pause = download time – play time + 10% of play time. For example, if your clip is 30 seconds long and it takes one minute to download, give your clip a 33-second buffer (60 seconds – 30 seconds + 3 seconds = 33 seconds).

Use clean video

The higher the quality of the original, the better the final result. Although frame rates and sizes of Internet video are usually smaller than those of television, computer monitors have much better color fidelity, saturation, sharpness, and resolution than conventional televisions. Even with a small window, image quality can be more important for digital video than for standard analog television. Artifacts and noise that are barely noticeable on TV can be obvious on a computer screen.

Remove noise and interlace

After you capture your video content, you might need to remove noise and interlacing.

Follow the same guidelines for audio

The same considerations exist for audio production as for video production. To achieve good audio compression, begin with clean audio. If you are encoding material from a CD, try to record the file using direct digital transfer instead of through the analog input of your sound card. The sound card introduces an unnecessary digital-to-analog and analog-to-digital conversion that can create noise in your source audio. Direct digital transfer tools are available for Windows and Macintosh platforms. To record from an analog source, use the highest-quality sound card available.

See also

“Embedding video in a SWF file” on page 303

Encoding video

About Flash Video encoding methods

The following sections describe the encoding settings, crop-and-trim controls, and for users of either the Flash Video Encoder or FLV QuickTime Export plug-in, the cue point embedding controls.

Flash Video Import wizard

The Flash Video Import wizard encodes video clips into the Flash Video (FLV) format when you import them. However, you can only encode one video clip at a time, and the process of encoding can be both time- and computing-intensive.

For users who work extensively with video-based content, Flash includes the Flash Video Encoder and the QuickTime Exporter.

Flash Video Encoder

The Flash Video Encoder batches process video clips, allowing you to encode several clips at a time without having to interrupt your workflow. In addition to selecting encoding options for video and audio content, the Flash Video Encoder can also embed cue points into video clips you encode, and edit the video using crop-and-trim controls.

For more information, see the online help included with the Flash Video Encoder application.

FLV QuickTime Export plug-in

If you have Flash 8 or later and QuickTime 6.1.1 installed on your computer, you can use the FLV QuickTime Export plug-in to export FLV files from supported video-editing applications. You can then import these FLV files directly into Flash.

The FLV Export plug-in supports the following video-editing applications:

- Adobe After Effects (Windows and Macintosh)
- Apple FinalCut Pro (Macintosh)
- Apple QuickTime Pro (Windows and Macintosh)
- Avid Xpress DV (Windows and Macintosh)

Using the FLV QuickTime Export plug-in to export FLV files from either Flash Video Encoder or other video-editing applications significantly streamlines working with FLV files in your Flash documents. With the FLV Export plug-in, you can select encoding options for video and audio content as you export, including frame rate, bit rate, quality, and other options. You can import FLV files directly into Flash without re-encoding the video after import.

For a video tutorial about using Flash Video Encoder, see www.adobe.com/go/vid0138.

Select a video encoding profile

Encoding profiles are based on the Flash Player version you intend to publish content for, and the data rate at which you want your video content to be encoded. An encoding profile using Flash Player 8 uses the On2 VP6 video codec. An encoding profile using Flash Player 7 uses the Sorenson Spark video codec to encode the video.

1 In the Encoding panel, select an encoding profile from the Flash Video Encoding Profile pop-up menu.

2 Verify that the encoding profile you've selected is appropriate for your intended application.

3 Do one of the following:

- To further adjust the encoding settings, or to modify the video clip's size or playback length using the crop-and-trim controls, click the Video or Crop and Resize tabs.
- To encode the video, click Continue.

See also

"Comparing the On2 VP6 and Sorenson Spark video codecs" on page 310

Specify advanced encoding settings

1 To specify advanced encoding options, select the Video tab.

2 Select a video codec to encode your content with from the Video Codec pop-up menu. If you are authoring for Flash Player 6 or 7, choose the Sorenson Spark codec; if you are authoring for Flash Player 8 or later, choose the On2 VP6 codec.

3 Select a frame rate.

By default, Flash Video Encoder uses the same frame rate as the source video. Adobe® recommends using the default frame rate unless you are experienced with video encoding, and have a specific application that requires modifying the source video's frame rate. To alter the frame rate, understand how modifying the frame rate affects the video quality.

Note: *To embed video clips in the SWF file, the frame rate of the video clip must be the same as that used by the SWF file. To encode video using the same frame rate as the FLA file, use the Video settings in the Video Import wizard.*

4 Select the keyframe placement for the video. Keyframes are video frames that contain complete data. For example, if you specify a keyframe interval of 30, Flash Video Encoder encodes a complete frame every 30 frames in the video clip. For frames between keyframe intervals, Flash stores only the data that changes from the preceding frame.

By default, Flash Video Encoder places a keyframe every two seconds of playback time. For example, if the video you're encoding has a frame rate of 30 fps, a keyframe is inserted every 60 frames. In general, the default keyframe value provides a reasonable level of control when seeking within a video clip. To select a custom keyframe placement value, be aware that the smaller the keyframe interval, the larger the file size.

5 Specify the quality of the video from the Quality pop-up menu.

The quality setting determines the data rate (or bit rate) of the encoded video. The higher the data rate, the better the quality of the encoded video clip. To specify a quality setting, do one of the following:

- Select a preset quality setting to automatically select a Data Rate value.
- Select Custom and enter a value, in kilobits per second, in the Maximum Data Rate text field.

***Note:** If you find that the preset quality settings do not work with your source footage, try specifying a custom maximum data rate.*

6 To resize the video clip, do the following:

- Select the Resize video check box.
- (Optional) To keep the aspect ratio the same as the original video clip, select the Maintain Aspect Ratio check box.

***Note:** If you resize a video clip's frame size, and do not select the Maintain Aspect Ratio check box, the video might become distorted.*

- Specify values for Width and Height. You can specify a frame size in pixels or as a percentage of the original image size.

7 Do one of the following:

- To move to the next panel of the Video Import wizard, click Next.
- To further modify the video's encoding settings, select either the Cue Point or the Crop and Resize tab.

See also

"Import video with Flash Media Server or FVSS" on page 305

"Embedding video in a SWF file" on page 303

Specify advanced audio encoding settings

The audio encoding settings portion of the dialog box is automatically selected when an audio-only file is added to the encoding list. This portion of the dialog box is disabled when you encode video content without a combined audio track.

***Note:** Select an encoding profile from the Flash Video Encoding Profile pop-up menu, which sets a corresponding profile for the audio-only encoding format.*

1 To activate the advanced audio encoding options, select Audio. The default audio codec is mp3.

***Note:** If the source video file has no audio track, or if you are encoding MPEG-1 files on a Macintosh computer, the audio encoding settings portion of the dialog box is disabled.*

2 Select a data rate from the Data Rate pop-up menu.

The data rate is the bit rate of the mp3 audio stream. Better quality audio tracks, such as music and significant background noise, require a higher bit rate. Simple dialogue can be compressed to a much higher degree. Higher bit-rate settings (encoded at 80 Kbps or higher) are encoded in stereo, while lower bit-rate settings (encoded at 64 Kbps or lower) are encoded in mono.

3 To move to the next panel of the Video Import wizard, click Next. To further modify the video clip, select another of the Advanced Settings tabs.

Embed cue points

Cue points cause the video playback to trigger other actions within the presentation. For example, you can create a Flash presentation video playing in one area of the screen while text and graphics appear in another area. A cue point placed in the video triggers an update to the text and graphics, letting them remain relevant to the content of the video.

Each cue point consists of a name and the time at which it occurs. You specify cue point times in hour:minute:second:milisecond format; the default frame rate is 30 fps. You can specify cue point times with any frame rate, and also express them in milliseconds rather than frame numbers.

To define and embed cue points, use Flash Video Encoder or import a video clip using the Video Import wizard.

1 From the Encoding Profiles panel of the Video Import wizard, select a predefined encoding profile from the Flash Video encoding profile pop-up menu, or create your own custom encoding profile using the encoding options in the Video tab.

2 Click the Cue Points tab.

3 To locate a specific frame (point in the video) to embed a cue point, use the playback head. For greater precision, select the playback head, and use the left and right arrow keys to locate specific points in the video. To locate a specific frame, use the pointer to move the playback head to the point in the video to embed a cue point. The video preview window visually identifies points in the video at which to insert a cue point. To locate specific points in time at which to embed cue points, use the elapsed time counter (located beneath the video preview window).

4 When the playback head is positioned on a frame where you want to embed a cue point, click the Add Cue Point button.

Flash Video Encoder embeds a cue point on that frame of the video, and populates the cue point list with a placeholder for the name of the new cue point, and the elapsed time and video frame at which the cue point is located (this is the time during playback when the event is triggered). Flash Video Encoder also displays a pop-up menu that lets you select the type of cue point to embed.

A cue point marker appears on the slider control at the point where the cue point was embedded. To adjust the placement of the cue point, use the cue point marker.

5 Specify the type of cue point to embed:

- Event cue points trigger ActionScript methods when the cue point is reached, and synchronize the video playback to other events in the Flash presentation.
- Navigation cue points are used for navigation and seeking, and to trigger ActionScript methods when the cue point is reached. Embedding a navigation cue point inserts a keyframe at that point in the video clip.

6 Enter parameters for the selected cue point. Parameters are a set of key-value pairs that you can add to the cue point. The parameters are passed to the cue point event handler as members of the single parameter object.

7 (Optional) Save the cue points you’ve created so that you can apply them to other video clips. Click the Save Cue Points button (the disk icon) on the cue points tab, and save the file to a location on your computer..

See also

“Importing and modifying Flash Video files” on page 301

“Encoding video” on page 312

Crop and resize video

Flash Video Encoder provides the following editing options to crop and resize video clips before encoding them:

Cropping alters the dimensions of a video clip. You can eliminate areas of the video to emphasize a particular focal point within the frame such as highlighting a character by removing ancillary imagery or removing unwanted backdrops.

Trimming edits the beginning and ending points (the *in* and *outpoints*) of a video. For example, you can adjust the trim of a video clip to begin playback 30 seconds into the full clip, removing unwanted frames.

1 From the Encoding Profiles panel of the Video Import wizard, specify an encoding setting for the video clip.

2 Click the Crop and Resize tab.

3 Enter values for the right, left, top, and bottom edges to crop the video, or use the slider controls to visually adjust the dimensions of the video. Guides in the preview window indicate the cropped area.

4 To set the in and out points, drag the in and out point markers below the scrubber bar until you finish adjusting the video clip size. For greater precision, select the in and out point markers, and use the left and right arrow keys to locate specific points in the video.

The video preview window visually identifies beginning and ending frames where you can trim the video clip. To locate specific times where you can trim the video clip, use the elapsed time counter (located in the Trim section of the dialog box).

5 To preview the video, drag the play head over the scrubber bar to ensure that the video plays appropriately.

6 To further modify the video's encoding settings, select either the Cue Point or the Encoding tab, or to return to the main Flash Video Encoding dialog box, click OK.

***Note:** When you encode the video, the original source video clip is not changed. You can always re-encode a video clip and specify new settings if your initial attempt doesn't produce the results you want.*

See also

"Encoding video" on page 312

Working with Premier Pro and After Effects

Working with Adobe Premiere Pro and Adobe Flash

Adobe Premiere Pro is a professional tool for editing video. If you use Adobe Flash to design interactive content for websites or mobile devices, you can use Adobe Premiere Pro to edit the movies for those projects. Adobe Premiere Pro gives you professional tools for frame-accurate video editing, including tools for optimizing video files for playback on computer screens and mobile devices.

Adobe Flash CS3 Professional is a tool for incorporating video footage into presentations for the web and mobile devices. Adobe Flash offers technological and creative benefits that let you fuse video with data, graphics, sound, and interactive control. The Adobe Flash Video format lets you put video on a web page in a format that almost anyone can view.

If you use Adobe Premiere Pro to export Adobe Flash Video files, you can use Adobe Flash to embed the content into interactive websites or applications for mobile devices. Adobe Flash can import sequence markers you add in an Adobe Premiere Pro sequence as cue points that can trigger events you designate in Adobe Flash, on playback.

If you export video files in other standard formats, Adobe Flash can encode your videos within Flash applications, using the latest compression technologies to deliver the greatest quality possible at small file sizes.

Moving content between Adobe Premiere Pro and Adobe Flash

After you start and edit a video in Adobe Premiere Pro, you can add sequence markers to the timeline that serve as cue points in a Flash application. Then, you can export the movie directly into the Adobe Flash Video format (FLV). You can choose from several Adobe Media Encoder presets that balance file size against audio and video quality to achieve the bit rate needed for any target audience or device. If you export the movie with an alpha channel, the movie can be easily used as a layer in a Flash project.

You can then import this movie into Adobe Flash for use in an interactive website or mobile application. Flash will read sequence markers as cue points you can use to trigger events in the Flash composition. In Flash, you can also customize the interface that surrounds your video.

Alternatively, because Flash can be used to create animations, you can start a movie as a Flash project, export it as a QuickTime file, then import the QuickTime file into Adobe Premiere Pro for editing. In Adobe Premiere Pro, for example, you could add titles or mix the animation with other video sources.

Tips for creating Adobe Flash Video

Follow these guidelines to deliver the best possible Flash video:

Work with video in the native format of your project until your final output

If you convert a precompressed digital video format into another format such as FLV, the previous encoder can introduce video noise. The first compressor already applied its encoding algorithm to the video, reducing its quality, frame size, and rate. That compression may have also introduced digital artifacts or noise. This additional noise affects the final encoding process, and a higher data rate may be required to encode a good-quality file.

Strive for simplicity

Avoid elaborate transitions—they don't compress well and can make your final compressed video look "chunky" during the change. Hard cuts (as opposed to dissolves) are usually best. Eye-catching video sequences—for instance showing an object zooming from behind the first track, doing a "page peel," or wrapping around a ball and then flying off the screen—don't compress well and should be used sparingly.

Know your audience data rate

When you deliver video over the Internet, produce files at lower data rates. Users with fast Internet connections can view the files with little or no delay for loading, but dial-up users must wait for files to download. Make the clips short to keep the download times within acceptable limits for dial-up users.

Select the proper frame rate

Frame rate indicates frames per second (fps). If you have a higher data rate clip, a lower frame rate can improve playback through limited bandwidth. For example, if you are compressing a clip with little motion, cutting the frame rate in half probably saves you only 20% of the data rate. However, if you are compressing high-motion video, reducing the frame rate has a much greater effect on the data rate.

Because video looks much better at native frame rates, leave the frame rate high if your delivery channels and playback platforms allow. For web delivery, get this detail from your hosting service. For mobile devices, use the device-specific encoding presets, and the device emulator available through Adobe Media Encoder in Adobe Premiere Pro. If you need to reduce the frame rate, the best results come from dividing the frame rate by whole numbers.

***Note:** When you embed video clips in the SWF file, the frame rate of the video clip must be the same as the frame rate of the SWF file. To encode video using the frame rate of the FLA file, use the Advanced Video Encoding settings in the Flash Video Import wizard.*

Select a frame size that fits your data rate and frame aspect ratio

At a given data rate (connection speed), increasing the frame size decreases video quality. When you select the frame size for your encoding settings, consider frame rate, source material, and personal preferences. To prevent pillarboxing, it's important to choose a frame size of the same aspect ratio as that of your source footage. For example, you get pillarboxing if you encode NTSC footage to a PAL frame size.

Adobe Premiere Pro makes several Adobe Flash Video presets available through Adobe Media Encoder. These include preset frame sizes and frame rates for the different television standards at different data rates. Use the following list of common frame sizes (in pixels) as a guide, or experiment with the various Adobe Media Encoder presets to find the best setting for your project.

Modem NTSC 4 x 3 162 x 120

Modem PAL 4 x 3 160 x 120

T1/DSL/cable NTSC 4 x 3 648 x 480

T1/DSL/cable PAL 4 x 3 768 x 576

Stream for best performance

To eliminate download time, provide deep interactivity and navigation capabilities, or monitor quality of service, stream Adobe Flash Video files with the Flash Media Server or use the hosted service from one of Adobe's Flash Video Streaming Service partners available through the Adobe website. For more details on the difference between Progressive Download and Streaming with Flash Media Server, see "Delivering Flash Video: Understanding the Difference Between Progressive Download and Streaming Video" on the Flash Developer Center website.

Know progressive download times

Know how long it will take to download enough of your video so that it can play to the end without pausing to finish downloading. While the first part of your video clip downloads, you may want to display other content that disguises the download. For short clips, use the following formula: Pause = download time – play time + 10% of play time. For example, if your clip is 30 seconds long and it takes one minute to download, give your clip a 33-second buffer (60 seconds – 30 seconds + 3 seconds = 33 seconds).

Remove noise and interlacing

For the best encoding, you might need to remove noise and interlacing.

The higher the quality of the original, the better the final result. Although frame rates and sizes of Internet video are usually smaller than those of television, computer monitors have much better color fidelity, saturation, sharpness, and resolution than conventional televisions. Even with a small window, image quality can be more important for digital video than for standard analog television. Artifacts and noise that are barely noticeable on TV can be obvious on a computer screen.

Adobe Flash is intended for progressive display on computer screens and other devices, rather than on interlaced displays such as TVs. Interlaced footage viewed on a progressive display can exhibit alternating vertical lines in high-motion areas. Thus, all the Adobe Flash Video presets in the Adobe Media Encoder have deinterlacing turned on by default.

Follow the same guidelines for audio

The same considerations apply to audio production as to video production. To achieve good audio compression, begin with clean audio. If you are encoding material from a CD, try to record the file using direct digital transfer instead of through the analog input of your sound card. The sound card introduces an unnecessary digital-to-analog and analog-to-digital conversion that can create noise in your source audio. Direct digital transfer tools are available for Windows and Macintosh platforms. To record from an analog source, use the highest-quality sound card available.

Working with Flash and After Effects

If you use Adobe® Flash® to create video or animation, you can use After Effects® to edit and refine the video. For example, you can export Flash animations and applications as QuickTime movies or Flash® Video (FLV) files. You can then use After Effects to edit and refine the video.

If you use After Effects to edit and composite video, you can then use Flash to publish that video. You can also export an After Effects video as Flash content for further editing in Flash.

Exporting QuickTime video from Flash

If you create animations or applications with Flash, you can export them as QuickTime movies using the File > Export Movie command in Flash. For a Flash animation, you can optimize the video output for animation. For a Flash application, Flash renders video of the application as it runs, allowing the user to manipulate it. This lets you capture the branches or states of your application that you want to include in the video file.

Exporting Flash video (FLV) from After Effects

When you render finished video from After Effects, select FLV as the output format in the render queue to export directly to the Flash Video (FLV) format. This launches the Flash Video Encoder application, which provides size, compression, and other output options. Any After Effects markers are added to the FLV file as cue points.

You can then import the FLV file into Flash and publish it in a SWF file, which can be played by Flash Player.

Importing and publishing video in Flash

When you import a FLV file into Flash, you can use various techniques, such as scripting or Flash components, to control the visual interface that surrounds your video. For example, you might include playback controls or other graphics. You can also add graphic layers on top of the FLV file for composite effects.

Composite graphics, animation, and video

Flash and After Effects each include many capabilities that allow you to perform complex compositing of video and graphics. Which application you choose to use will depend on your personal preferences and the type of final output you want to create.

Flash is the more Web oriented of the two applications, with its small final file size. Flash also allows for runtime control of animation. After Effects is oriented towards video and film production, provides a wide range of visual effects, and is generally used to create video files as final output.

Both applications can be used to create original graphics and animation. Both use a timeline and offer scripting capabilities for controlling animation programmatically. After Effects includes a larger set of effects, while Flash's ActionScript™ language is the more robust of the two scripting environments.

Both applications allow you to place graphics and effects on separate layers for compositing. These layers can be turned on and off as needed.

In Flash, composites do not affect the video content directly; they affect only the appearance of the video during playback in Flash Player. In contrast, when you composite with imported video in After Effects, the video file you export actually incorporates the composited graphics and effects.

Because all drawing and painting in After Effects is done on layers separate from any imported video, it is always non-destructive. Flash has both destructive and non-destructive drawing modes.

Exporting After Effects content for use in Flash

You can export After Effects content for use in Flash. You can export a SWF file that can be played immediately in Flash Player or used as part of another Flash project. When you export content from After Effects in SWF format, the content is flattened and rasterized in the SWF file.

Importing Flash SWF files into After Effects

Flash has a unique set of vector art tools that make it useful for a variety of drawing not possible in After Effects or Adobe® Illustrator®. You can import SWF files into After Effects to composite them with other video or render them as video with additional creative effects. When After Effects imports a SWF file, its internal keyframes are preserved so that you can continue to use them for timing other effects.

The Continuously Rasterized SWF Import feature in After Effects allows you to bring SWF file content into After Effects as flattened art, with support for alpha channels. Because the rasterization is continuous, vector art in the SWF file that is scaled scales smoothly in After Effects. This import method allows you to use the root layer or object of your SWF files as a smoothly rendered element in After Effects, allowing the best capabilities of each tool to work together.

Using ActionScript to play external Flash Video

Playing back external FLV files dynamically

An alternative to importing video into the Flash authoring environment it to use either the FLVPlayback component or ActionScript to dynamically play external FLV files in Flash Player. You can play FLV files posted as HTTP downloads or as local media files.

Create FLV files by importing video into the Flash authoring tool and exporting it as an FLV file. If you have Macromedia Flash Professional 8 or Flash CS3 Professional, you can use the FLV Export plug-in to export FLV files from supported video-editing applications.

To play back an external FLV file, post an FLV file to a URL (either an HTTP site or a local folder) and add either the FLVPlayback component or ActionScript code to the Flash document to access the file and control playback during runtime.

Using external FLV files provides the following capabilities that are not available when using imported video:

- You can use longer video clips without slowing down playback. External FLV files are played using *cached memory*, which means that large files are stored in small pieces and accessed dynamically; they do not require as much memory as embedded video files.
- An external FLV file can have a different frame rate from the Flash document in which it plays. For example, you can set the Flash document frame rate to 30 fps and the video frame rate to 21 fps, which gives you greater control in ensuring smooth video playback.
- With external FLV files, Flash document playback does not have to be interrupted while the video file is loading. Imported video files can sometimes interrupt document playback to perform certain functions (for example, to access a CD-ROM drive). FLV files can perform functions independently of the Flash document, and so do not interrupt playback.
- Captioning video content is easier with external FLV files because you can use callback functions to access metadata for the video.

For more information on playing back FLV files, see “Playing back external FLV files dynamically” in *Learning ActionScript 2.0 in Adobe Flash* or “Basics of video” in *Programming ActionScript 3.0*.

See also

“Export a Flash video file” on page 451

“About Flash Video encoding methods” on page 312

Behaviors used in video playback

Video behaviors provide one way to control video playback. Behaviors are prewritten ActionScript scripts that you add to a triggering object to control another object. Behaviors add the power, control, and flexibility of ActionScript coding to your document without having to create the ActionScript code. Video behaviors play, stop, pause, rewind, fast-forward, show, and hide a video clip.

To control a video clip with a behavior, use the Behaviors panel to apply the behavior to a triggering object, such as a movie clip. Specify the event that triggers the behavior (such as releasing the movie clip), select a target object (the video that is affected by the behavior), and when necessary, select settings for the behavior, such as the number of frames to rewind.

***Note:** The triggering object must be a movie clip. You cannot attach video playback behaviors to button symbols or button components.*

The following behaviors in Flash control embedded video:

Behavior	Purpose	Parameters
Play Video	Plays a video in the current document.	Instance name of target video
Stop Video	Stops the video.	Instance name of target video
Pause Video	Pauses the video.	Instance name of target video
Rewind Video	Rewinds the video by the specified number of frames.	Instance name of target video Number of frames

Behavior	Purpose	Parameters
Fast Forward Video	Fast-forwards the video by the specified number of frames.	Instance name of target video Number of frames
Hide Video	Hides the video.	Instance name of target video
Show Video	Shows the video.	Instance name of target video

Control video playback using behaviors

1 Select the movie clip to trigger the behavior.

2 In the Behaviors panel (Window > Behaviors), click the Add (+) button, and select the desired behavior from the Embedded Video submenu.

3 Select the video to control.

4 Select a Relative or Absolute path.

5 If required, select settings for the behavior parameters and click OK.

6 In the Behaviors panel under Event, click On Release (the default event) and select a mouse event. To use the On Release event, leave the option unchanged.

See also

"Using absolute and relative target paths" on page 72

"Controlling video playback using the Timeline" on page 301

About the FLVPlayback component

The FLVPlayback component does the following:

- Provides a set of prefabricated skins to customize playback controls and the look and feel of the user interface.
- Lets advanced users create their own custom skins.
- Provides cue points to synchronize your video with the animation, text, and graphics in your Flash application.
- Provides live preview of customizations.
- Maintains a reasonably sized SWF file for easy download.

The FLVPlayback component is the display area in which you view video. The FLVPlayback component includes the FLV Custom UI controls, a set of control buttons that play, stop, pause, and control playback the video.

Configure the FLVPlayback component

1 With the component selected, open the Property inspector (Window > Properties > Properties) and enter an instance name.

2 Select Parameters in the Property inspector or open the Component inspector (Window > Components).

3 Enter values for parameters or use default settings.

For each FLVPlayback component instance you can set the following parameters in the Property inspector or in the Component inspector:

***Note:** In most instances, it is not necessary to alter the settings in the FLVPlayback component unless you want to change the appearance of a video skin. The Video Import wizard sufficiently configures the parameters for most deployments.*

autoPlay Boolean value that determines how to play the FLV. If `true`, the FLV plays immediately when it is loaded. If `false`, loads the first frame and pauses. The default value is `true`.

autoRewind Boolean value that determines whether the FLV is automatically rewound. If `true`, the Video component automatically rewinds the FLV to the beginning when the playhead reaches the end or when the user clicks the stop button. If `false`, the Video component does not automatically rewind the FLV. The default value is `true` .

autoSize Boolean value that, if `true`, resizes the component at runtime to use the source FLV dimensions. The default value is `false`.

***Note:** The encoded frame size of the FLV is not the same as the default dimensions of the FLVPlayback component.*

bufferTime Number of seconds to buffer before beginning playback. The default value is 0.

contentPath String that specifies the URL to an FLV or to an XML file that describes how to play the FLV. Double-click the Value cell for this parameter to activate the Content Path dialog box. The default is an empty string. If you do not specify a value for the `contentPath` parameter, nothing happens when Flash executes the FLVPlayback instance.

isLive Boolean value that, if `true`, specifies that the FLV is streaming live from FMS. The default value is `false`.

cuePoints A string that specifies the cue points for the FLV. Cue points allow you to synchronize specific points in the FLV with Flash animation, graphics, or text. The default value is an empty string.

maintainAspectRatio A Boolean value that, if `true`, resizes the video player in the FLVPlayback component to retain the source FLV aspect ratio; the source FLV is still scaled and the FLVPlayback component itself is not resized. The `autoSize` parameter takes precedence over this parameter. The default value is `true`.

skin A parameter that opens the Select Skin dialog box and allows you to choose a skin for the component. The default value is None. If you choose None, the FLVPlayback instance does not have control elements that allow the user to play, stop, or rewind the FLV, or take other actions that the controls make possible. If the `autoPlay` parameter is set to true, the FLV plays automatically. For more information, see “Customizing the FLVPlayback component” in *Using ActionScript 3.0 Components* or *ActionScript 2.0 Components Language Reference*.

totalTime Total number of seconds in the source FLV. The default value is 0. If you use progressive download, Flash uses this number if it is set to a value greater than zero (0). Otherwise, Flash tries to take the time from metadata.

***Note:** If you're using FMS or FVSS, this value is ignored; the total time of the FLV is taken from the server.*

volume A number from 0 to 100 that represents the percentage of maximum volume at which to set the volume.

Specify the contentPath parameter

If you imported a local video clip into Flash for use with progressively downloaded or streaming video content, update the `contentPath` parameter of the FLVPlayback component before uploading your content to a web server. The `contentPath` parameter specifies the name and location of the FLV on the server, and implies the playback method (for example, progressively download using HTTP, or streaming from Flash Media Server using RTMP).

1 With the component selected, open the Property inspector (Window > Properties > Properties) and select Parameters in the Property inspector, or open the Component inspector (Window > Component Inspector).

2 Enter values for parameters, or use the default settings as appropriate. For the `contentPath` parameter, do the following:

a Double-click the Value cell for the `contentPath` parameter to activate the Content Path dialog box.

b Enter the URL or local path to either the FLV file or the XML file (for Flash Media Server or FVSS) that describes how to play the FLV.

If you do not know the location of the FLV or XML file, click the folder icon to navigate to the correct location. When browsing for an FLV file, if it is at or below the location of the target SWF file, Flash automatically makes the path relative to that location so that it is ready for serving from a web server. Otherwise, it is an absolute Windows or Macintosh file path.

If you specify an HTTP URL, the FLV file is a progressive download FLV file. If you specify a URL that is a Real-Time Messaging Protocol (RTMP) URL, the FLV streams from a Flash Media Server (FMS). A URL to an XML file could also be a streaming FLV file from FMS or from a FVSS.

***Note:** When you click OK on the Content Path dialog box, Flash updates the value of the* `cuePoints` *parameter, too, because you might have changed the* `contentPath` *parameter so that the* `cuePoints` *parameter no longer applies to the current content path. As a result, you lose any disabled cue points, although not ActionScript cue points. For this reason, you might want to disable non-ActionScript cue points through ActionScript, rather than through the Cue Points dialog box.*

You can also specify the location of an XML file that describes how to play multiple FLV streams for multiple bandwidths. The XML file uses Synchronized Multimedia Integration Language (SMIL) to describe the FLV files. For a description of the XML SMIL file, see “Using a SMIL file” in the *ActionScript 2.0 Components Language Reference*.

Media components (Flash Player 6 and 7)

***Note:** The media components were introduced in Macromedia Flash MX Professional 2004. If you are developing content to use with Macromedia Flash Player 8, instead use the FLVPlayback component introduced in Macromedia Flash Professional 8. The FLVPlayback component provides improved functionality, giving you more control over video and audio playback in the Flash environment.*

The media component suite consists of three components: MediaDisplay, MediaController, and MediaPlayback. With the MediaDisplay component, to add media to your Flash documents, drag the component to the Stage and configure it in the Component inspector. In addition to setting the parameters in the Component inspector, you can add cue points to trigger other actions. The MediaDisplay component has no visual representation during playback; only the video clip is visible.

The MediaController component provides user interface controls that let the user interact with streaming media. The Controller features Play, Pause, and Rewind to Start buttons and a volume control. It also includes playbars that show how much of the media has loaded and how much has played. A playhead slider can be dragged forward and backward on the playbar to navigate quickly to different parts of the video. Using behaviors or ActionScript, you can easily link this component to the MediaDisplay component to show streaming video and provide user control.

The MediaPlayback component provides the easiest and quickest way to add video and a controller to your Flash documents. The MediaPlayback component combines the MediaDisplay and MediaController components into a single, integrated component. The MediaDisplay and MediaController component instances are automatically linked to each other for playback control.

To configure parameters for playback, size, and layout for all three components, use the Component inspector or the Parameters tab in the Property inspector. All the media components work equally well with mp3 audio content.

For more information on the media components, “Media components,” in the *ActionScript 2.0 Components Language Reference*.

Chapter 16: Creating e-learning content

Adobe® Flash® CS3 Professional e-learning content is designed to facilitate rapid e-learning development, letting you create online courses and instructional materials.

***Note:** The Flash e-learning content works only with ActionScript™ 2.0 documents. E-learning content will not work with ActionScript 3.0*

Getting started with Flash e-learning

Flash e-learning overview

Flash learning interactions help you create interactive online (e-learning) courses that run in Flash. Following are some of the benefits:

- Anyone with a Flash-enabled web browser can use the instructional content.
- You can customize the interface to meet your needs and create high-quality interfaces that load quickly and look the same on different platforms.
- Interactions added to your online course provide a simple interface for entering data without writing code.
- Each Flash learning interaction can send tracking information to a server-side learning management system (LMS) that complies with the Aviation Industry CBT Committee (AICC) protocol or Shareable Content Object Reference Model (SCORM) standards.
- The quiz templates track cumulative results from a sequence of interactions and can route them to the LMS with an enhanced data tracking functionality that is either AICC or SCORM compliant.

Requirements

Your e-learning courseware runs on any computer with Flash® Player 6 or later and a Flash-enabled web browser. The e-learning content is compatible only with ActionScript 2.0, so you must specify ActionScript 2.0 when publishing your Flash documents using the e-learning courseware.

To track user data from the Flash learning interactions, you must have the following:

- A web server-side LMS, such as an AICC- or SCORM-compatible system
- Internet Explorer 4.0 or Netscape Navigator 4.0 or later (Windows), or Netscape 4.5 or later (Macintosh). Tracking to an LMS with learning interactions does not work with Internet Explorer on the Macintosh.

Flash learning interactions

An interaction is a part of a Flash application in which the user interacts with the application to provide a response. A typical response might be answering a question, selecting True or False, or clicking an area of the screen. Use the following six learning interactions to build interactive courseware:

True Or False The user chooses either true or false.

Multiple Choice The user selects among multiple answers.

Fill In The Blank The user types a response that is checked against matching phrases.

Drag And Drop The user responds to a question by dragging one or more onscreen objects to a target.

Hot Spot The user responds by clicking a region (or regions) on the screen.

Hot Object The user responds by clicking an object (or objects) on the screen.

Each learning interaction has unique parameters that determine how the interaction appears to the user. For additional information about Flash components, see *Using ActionScript 2.0 Components* or *Using ActionScript 3.0 Components*.

Including a Flash learning interaction in a document

Quiz templates and stand-alone interactions

Use either quiz templates or stand-alone interactions in your Flash documents:

- The quiz templates are designed for scenarios in which interaction-based quizzes are required, or tracking is necessary. The quiz learning interactions are graphically designed to fit into the quiz format. The quiz templates contain a mechanism that counts a cumulative score and starts and stops the necessary tracking in both AICC- and SCORM-compliant APIs.
- The stand-alone interactions are designed for scenarios that require a single interaction, or a series of interactions that need to fit into a specific layout within a Flash document. These interactions are available from the common library and are graphically designed for stand-alone use. You can track the results for each stand-alone interaction and submit them to an AICC-compatible LMS.

To initialize SCORM tracking, use a quiz template.

Create a quiz with a quiz template

Each of the three quiz templates that come with Flash has a different graphical look and feel, but they are otherwise identical. Each template contains the following elements:

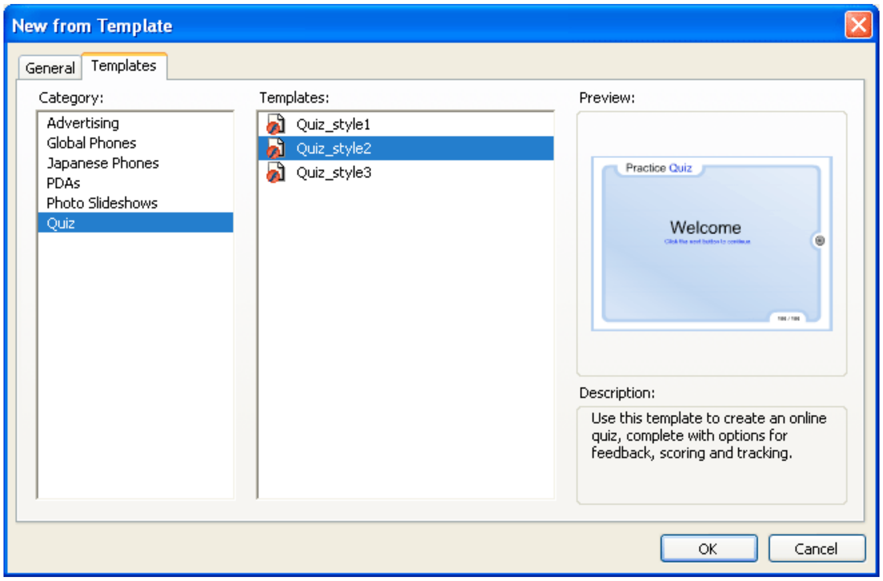
- A Welcome page
- One of each of the six learning interaction types
- A Results page
- Navigation elements
- ActionScript to gather AICC and SCORM tracking information

The quiz templates are fully functional. After creating a document from a quiz template, you can immediately test the document, before modification, to see how the quiz functions.

1 Create a file (File > New).

2 In the New From Template window, select the Templates tab.

3 In the Category column, select Quiz; in the Templates column, select one of the quiz styles.

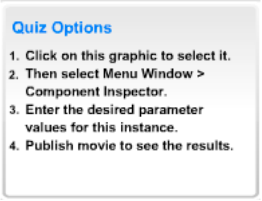


Use this template to create an online quiz, complete with options for feedback, scoring and tracking.

Set quiz parameters

Quiz parameters control how the entire quiz is presented to users—for example, whether the questions are presented in a random or sequential order, the number of questions to display, and whether the Results page appears.

1 Select the Quiz Options component with instructions to the left of the Stage in the quiz template. This component lets you set the parameters for the quiz.



Quiz Options

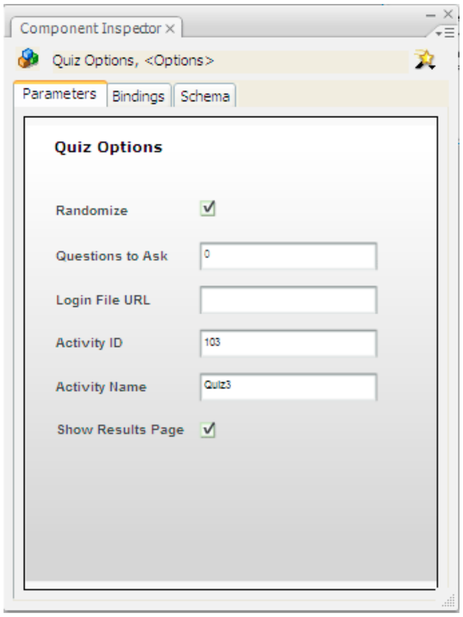
1. Click on this graphic to select it.
2. Then select Menu Window > Component Inspector.
3. Enter the desired parameter values for this instance.
4. Publish movie to see the results.

***Note:** These instructions do not appear in the SWF file.*

2 Do one of the following to open the Component inspector:

- Select Window > Component Inspector.

- In the Property inspector, click Launch Component Inspector.



Quiz Options

Randomize ☑

Questions to Ask 0

Login File URL

Activity ID 103

Activity Name Quiz3

Show Results Page ☑

Note: If the text in the Component inspector is too small to be legible, undock the panel and drag a corner of the panel to enlarge it.

3 Select Randomize to present the quiz questions in a random order.

4 In the Questions To Ask box, specify the number of questions to ask for one presentation of the quiz. If you set this number to 0, the quiz uses all the questions you add to the document. If you enter a number larger than the number of questions in the quiz, the quiz displays only the number of questions that are in the quiz and does not duplicate any of them.

For example, if you have 10 interactions in your quiz, you can specify that a lesser number, such as 5 interactions, appear to the user. Use this feature with the Randomize feature to create quizzes with unexpected questions in an unexpected order.

5 Enter the URL to redirect the user.

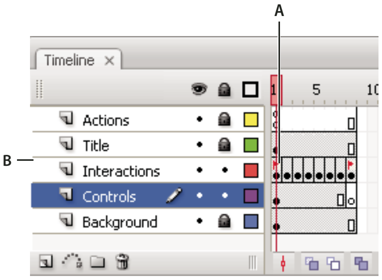
When an AICC-compliant LMS starts a quiz, it includes parameters that the HTML code looks for when it executes the `embed` tag for the Flash application, and the course loads properly. If no parameters are specified, the user is redirected to the URL specified in the Login File URL field. If this field is blank or the Flash file was published with the SCORM template, the redirection does not occur.

6 In the Activity ID and Activity Name boxes, enter the activity ID and activity name of your LMS. If you are not using an LMS, either accept or delete the default entries.

7 Select Show Results Page to present quiz results to users after they complete the quiz.

Modify learning interactions in a quiz template

Each question in the quiz is considered an interaction. When you use a quiz template, you place interactions sequentially between the first and last frame of the Interactions layer on the root Timeline. Add or remove frames and keyframes as needed, as long as the interactions remain sequential and the first and last frames are reserved for the Welcome and Results pages. The number of frames between the Welcome and Results page keyframes are used to calculate the score.



A. First frame of the layer *B. Interactions layer*

For example, the following 12 keyframes on the Interactions layer comprise a 10-question quiz:

- Frame 1 = Welcome page keyframe
- Frames 2–11 = Interactions keyframes
- Frame 12 = Results page keyframe

1 Select the first frame on the Interactions layer and modify the text of the Welcome page. Include text to indicate that the user must click Next to continue. Do not add an interaction to this page.

2 Select each of the learning interactions in the next six frames and do one of the following:

- To use the interaction, configure it.
- If you do not want to use the interaction, remove it.

3 Select the last frame in the Interactions layer and modify the text of the Results page. Leave the supplied dynamic text field names intact, or the results will not appear. Do not delete or place interactions in this frame. If the Results Page quiz parameter is turned off for the quiz, this frame is not called, but it is still reserved.

Configure a Learning Interaction component

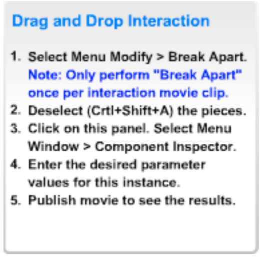
Included with each quiz template is one of each of the six learning interaction types, stored in movie clips in the library. These movie clips are containers for the collection of elements that make up each interaction. When you add an interaction (movie clip) to the Stage, break it apart to edit the individual objects.

1 Select the entire learning interaction, and then select Modify > Break Apart. This action breaks the interaction into individual objects that you can modify.

Note: *Break apart the interaction only once.*

2 Deselect all the items on the Stage (Control+Shift+A).

3 Select the Learning Interaction component.

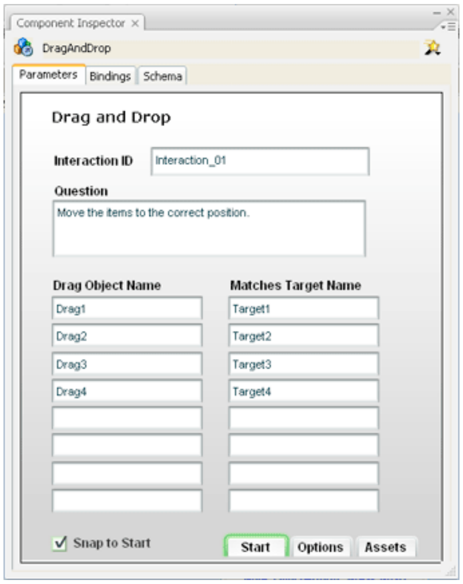


Drag and Drop Interaction

1. Select Menu Modify > Break Apart.
   Note: Only perform "Break Apart" once per interaction movie clip.
2. Deselect (Crtl+Shift+A) the pieces.
3. Click on this panel. Select Menu Window > Component Inspector.
4. Enter the desired parameter values for this instance.
5. Publish movie to see the results.

***Note:** Do not delete these instructions; they contain necessary ActionScript code and do not appear in the SWF file.*

4 In the Property inspector, click Launch Component Inspector.



Drag and Drop

Interaction ID: Interaction_01

Question: Move the items to the correct position.

Drag Object Name	Matches Target Name
Drag1	Target1
Drag2	Target2
Drag3	Target3
Drag4	Target4

☑ Snap to Start

5 If the Flash application sends tracking information to a server-side LMS, specify a name for the interaction in the Interaction ID box. Each interaction in the quiz templates is uniquely named. If you add interactions from the library or you are not using the quiz template, follow your LMS specifications to uniquely name each interaction in your file.

6 In the Question box, type the text the user will see. This text can be a question or instructions for the user.

7 Configure the learning interaction.

8 At the bottom of the Component inspector, click Options and enter feedback and Knowledge Track parameters for the learning interaction.

***Note:** Documents created using a quiz template have the Knowledge Track option turned on and the Navigation option turned off (the default settings) for each learning interaction, because the quiz template has its own navigation controls.*

9 (Optional) Click the Assets button, and change the assets for the learning interaction.

See also

“Configure a Drag and Drop interaction” on page 338

“Configure a Fill In The Blank interaction” on page 339

"Configure a Hot Object interaction" on page 340

"Configure a Hot Spot interaction" on page 341

"Configure a Multiple Choice interaction" on page 342

"Configure a True or False interaction" on page 343

"Adding, naming, and registering assets" on page 334

"Set Knowledge Track options for a learning interaction" on page 344

"Set navigation options for a learning interaction" on page 344

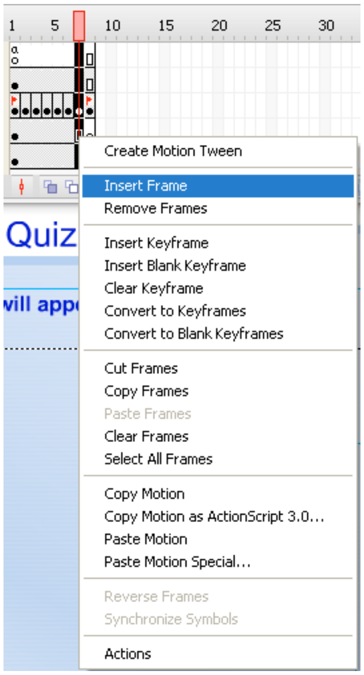
Add learning interactions to a quiz template

When you use a quiz template, you add learning interactions to the Interactions layer.

1 In the first layer of the Timeline, select the frame that precedes the frame number where you want to add the interaction. For example, to add an interaction to Frame 8, select Frame 7.

2 Shift-click the same frame number on the other layers to also select those frames.

3 Right-click (Windows) or Control-click (Macintosh) a selected frame and select Insert Frames to extend the Timeline evenly across all layers.



4 On the Interactions layer, select the frame you added and select Insert > Timeline > Blank Keyframe.

5 To add an interaction, do one of the following:

- To copy and paste an interaction that already exists in the Timeline, right-click (Windows) or Control-click (Macintosh) the keyframe with the interaction and select Copy Frames. Paste the frame in the blank keyframe that you inserted in step 4. In this copy of the interaction, modify objects on the Stage or the settings in the Component inspector.

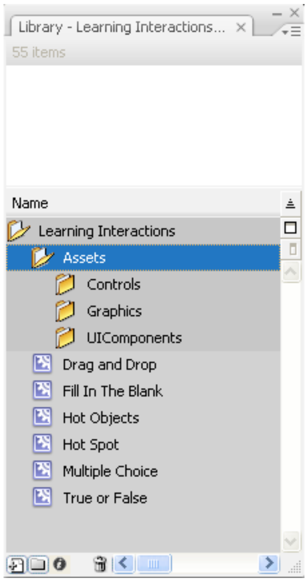
- To use an interaction from the library, drag the desired interaction movie clip type from the Learning Interactions library (Window > Common Libraries > Learning Interactions) to the blank keyframe. Break the interaction apart (select the interaction and select Modify > Break Apart), and edit the assets and parameters.

Add learning interactions to a document (no quiz template)

To add learning interactions to a Flash document that does not use a quiz template, place stand-alone learning interactions in the Timeline in a single frame, sequential frames (for example, 10 questions in 10 sequential frames), or labeled frames.

1 Select the appropriate layer and select Insert > Timeline > Blank Keyframe.

2 Select Window > Common Libraries > Learning Interactions.



The Learning Interactions library includes six types of learning interaction movie clips: Drag And Drop, Fill In The Blank, Hot Objects, Hot Spot, Multiple Choice, and True Or False. In addition, the library contains a folder called Assets that contains subfolders called Controls, Graphics, and UIComponents. These folders are used for customizing learning interactions.

3 Select the new keyframe and drag one of the Learning Interaction movie clips from the Library panel to the Stage.

4 Reposition the interaction by dragging it to where you want it to appear on the Stage.

5 Configure the learning interaction.

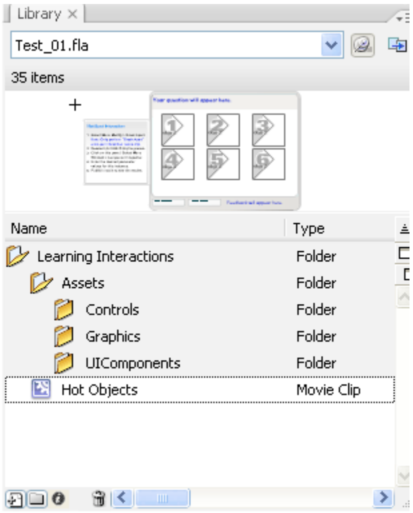
Note: *Watch the frame count across layers as you add and remove keyframes. All layers must end at the same frame number along the Timeline so that the frame count is the same in all layers.*

See also

“Changing buttons, check boxes, and radio buttons” on page 346

Managing library assets for learning interactions

When you drag a learning interaction from the Learning Interactions common library to the Stage, the symbols that comprise the learning interaction are copied from the common library to the library of the Flash document you are creating. For example, if you copy a Hot Object learning interaction from the Learning Interactions common library to your document, the symbols in the following example become part of the document library.



Test_01.fla

35 items

Name	Type
Learning Interactions	Folder
Assets	Folder
Controls	Folder
Graphics	Folder
UIComponents	Folder
Hot Objects	Movie Clip

If you're using a quiz template, the learning interaction symbols are already included in your document library.

To manage library assets, create folders for each graphical interaction, place the folders in the Assets folder, and keep the movie clips associated with the interaction in the new folder.

Remove a learning interaction from the Timeline

When you remove a learning interaction from the Timeline, maintain the sequence of learning interactions. If you remove a frame from the Interactions layer, you also need to remove it from all other layers.

1 On the Interactions layer, select the keyframe containing the interaction to delete. Shift-select the same frame number on other layers to also delete those frames.

2 To delete frames across all layers, do one of the following:

- Right-click (Windows) or Control-click (Macintosh) the keyframe and select Remove Frames.
- Select Edit > Timeline > Remove Frames.

***Note:** Watch the frame count across layers as you add and remove keyframes. All layers must end at the same frame number along the Timeline so that the frame count is the same in all layers.*

Check if a movie clip is broken apart

❖ Select a text field or any other single element of the learning interaction on the Stage.

If a grouped object is selected, the interaction is not broken apart.

If you can select a single text field or another element, the interaction is broken apart and you can proceed with editing.

Test a quiz

Test a quiz frequently as you add and remove interactions.

1 Select Control > Test Movie.

2 Answer the questions as they appear.

3 When you complete the quiz, close it in the Flash Player window to return to the workspace.

Adding, naming, and registering assets

Adding, naming, and registering assets and Learning Interaction component instances

Each Flash learning interaction consists of the following assets:

- An interaction component
- Dynamic text fields
- Distractor elements
- User interface (UI) components

Every interaction has an interaction component associated with it to configure its unique parameters. These components do not need to be named.

The collection of assets for each interaction type is stored in movie clip symbols in the library. These movie clips provide mobility for the assets so they can be copied to keyframes or among files. The movie clips are only containers and are not necessary to make the interaction work.

You do not need to use the movie clip containers or the templates—instead, you can add your own assets to the Stage, add a Learning Interaction component to the Stage, and then register the assets' instance names in the Component inspector for the interaction.

Remember the following aspects about naming assets:

- Interaction components do not need to be named.
- UI components need to have unique names for similar interaction types.
- Each graphic distractor (Drag object, Target object, hot spot, and hot object) must have a unique instance name.
- Text fields can share the same instance names across multiple interactions.

After you name the assets on the Stage, register those names in the Component inspector for the learning interaction, so that the scripts can control the assets.

Name UI components (RadioButton, CheckBox, Button, and TextInput)

When you use similar interaction types, give each UI component a unique name. For example, if you create two Multiple Choice interactions, the second interaction requires unique instance names for the CheckBox and the Button components. Register these new instance names in the Component inspector for the learning interaction.

1 Select the UI component instance on the Stage.

2 In the Property inspector, type a name in the Instance Name box.

3 Register the name in the Component inspector for the interaction.

Name dynamic text fields

If you have more than one of any type of learning interaction in a quiz, the objects in each learning interaction must have unique names. Register these new unique instance names in the Component inspector for the learning interaction.

1 Select the dynamic text field on the Stage.

2 In the Property inspector, type a name in the Instance Name box.

***Note:** Enter the instance name, not the variable name.*

3 Register the name in the Component inspector.

Register dynamic text fields and UI components

1 Select the Learning Interaction component (to the left of the Stage in the quiz template), and open the Component inspector, if necessary, from the Property inspector.

2 Click Assets at the bottom of the panel.

3 Enter the name in the appropriate instance name box.

Name and register graphic distractors

Graphic distractors such as Drag objects, Target objects, hot spots, and hot objects must be named uniquely across all interactions. In a file with two Drag and Drop interactions, each containing four Drag objects, each of the eight Drag objects in the file must be named uniquely. A sequential naming scheme is usually the easiest to work with—for example, the Drag objects in the first interaction could be named Drag 1, Drag 2, Drag 3, and Drag 4, and the Drag objects in the second interaction could be named Drag A, Drag B, Drag C, and Drag D. This system ensures that the scripts work properly and the interactions behave as intended.

Name graphic distractors

1 Check that the objects on the Stage are instances of learning interactions or movie clip symbols.

2 Select an object on the Stage.

3 In the Property inspector, type a name in the Instance Name box.

4 Repeat steps 1 to 3 for each object on the Stage.

5 Register the names.

Register a distractor instance name

1 Select the Learning Interaction component (to the left of the Stage in the quiz template), and open the Component inspector from the Property inspector.

2 Enter the name in the Component inspector, under Instance Name.

Text field names

Text fields can share the same names from interaction to interaction. The question text field in interaction 1 can be named the same as the question text field in interaction 2, and so on. Register these and all asset names with the interaction components.

Asset name defaults

The assets supplied in the movie clip interaction containers are prenamed with the instance names listed in the following tables.

Drag and Drop learning interaction asset names

Asset	Description	Object type	Instance name
Question text field	Holds question text	Dynamic text field	Template_Question
Feedback text field	Holds feedback text	Dynamic text field	Template_Feedback
Control button	Submits user response and controls navigation	Flash UI Button component	Template_ControlButton
Reset button	Resets Drag objects	Flash UI Button component	Template_ResetButton
1-8 Drag objects	Drag object distractors	Movie clip symbol	Drag1 – Drag8
1-8 Target objects	Targets for Drag objects	Movie clip symbol	Target1 – Target8

Fill In The Blank learning interaction asset names

Asset	Description	Object type	Instance name
Question text field	Holds question text	Dynamic text field	Template_Question
Feedback text field	Holds feedback text	Dynamic text field	Template_Feedback
User entry field	User types answer into this text field	Flash UI TextInput component	Template_UserEntry
Control button	Submits user response and controls navigation	Flash UI Button component	Template_ControlButton

Hot Object learning interaction asset names

Asset	Description	Object type	Instance name
Question text field	Holds question text	Dynamic text field	Template_Question
Feedback text field	Holds feedback text	Dynamic text field	Template_Feedback
Control button	Submits user response and controls navigation	Flash UI Button component	Template_ControlButton
Reset button	Resets hot object distractors	Flash UI Button component	Template_ResetButton
1-8 hot objects	Hot object distractors	Movie clip symbol	HotObject1 - 8

Hot Spot learning interaction asset names

Asset	Description	Object type	Instance name
Question text field	Holds question text	Dynamic text field	Template_Question
Feedback text field	Holds feedback text	Dynamic text field	Template_Feedback

Asset	Description	Object type	Instance name
Control button	Submits user response and controls navigation	Flash UI Button component	Template_ControlButton
Reset button	Resets hot spot distractors	Flash UI Button component	Template_ResetButton
1-8 hot spots	Hot spot distractors	Movie clip symbol	HotSpot1 - 8

Multiple Choice learning interaction asset names

Asset	Description	Object type	Instance name
Question text field	Holds question text	Dynamic text field	Template_Question
Feedback text field	Holds feedback text	Dynamic text field	Template_Feedback
Control button	Submits user response and controls navigation	Flash UI Button component	Template_ControlButton
3-8 check boxes	Check box distractors	Flash UI CheckBox component	Checkbox1-8

True or False learning interaction asset names

Asset	Description	Object type	Instance name
Question text field	Holds question text	Dynamic text field	Template_Question
Feedback text field	Holds feedback text	Dynamic text field	Template_Feedback
Control button	Submits user response and controls navigation	Flash UI Button component	Template_ControlButton
2 radio buttons	True or false radio button distractors	Flash UI RadioButton component	Template_Radio1, Template_Radio2

Configuring learning interactions

About distractors

For each of the six interactions, you must enter specific parameters for the quiz to function properly. A Drag and Drop interaction requires that you specify the Target object and the Drag object. Each Target object and Drag object is referred to as a *distractor*. A distractor is one of a series of selectable choices. This term is used for the choices in each of the learning interactions. For example, with a Multiple Choice learning interaction, you enter the multiple-choice distractors.

Configure a Drag and Drop interaction

As many as eight Drag objects and eight Target objects can be in each Drag and Drop interaction. Each Drag object can snap to any target named in the Drag and Drop component for evaluation. Drag objects can also share targets; for example, both Drag 1 and Drag 2 can match Target 8. You can also specify a target without matching a Drag object to it, which lets you add incorrect target distractors for evaluation.

1 If you are not using a quiz template, place the learning interaction on the Stage. If you are using a quiz template, select the frame on the Interactions layer that contains the Drag and Drop interaction (Frame 2, if you did not add or remove keyframes).

2 Break the movie clip apart (Modify > Break Apart), display the Component inspector, and type the interaction ID and the question.

3 In the Drag Object Name column, list the instance names for the Drag objects on the Stage. Each Drag object must have a unique name. If you add a new Drag object on the Stage, enter its name here.

4 In the Matches Target Name column, list the matching target instance name for that Drag object. Each target must have a unique name. If you add a new target on the Stage, enter its name here. If you enter a Drag instance name in the Drag Object Name column, enter a corresponding Target instance name in the Matches Target Name column. You can enter a Target instance name in the Matches Target Name column without a matching Drag instance name, which adds a target that can be snapped to but is not evaluated as a correct match.

5 Select Snap to Start to make the Drag objects snap back to their original position if they do not snap to a registered target.

6 Select each instance of the Drag object or Target object on the Stage. Use the Property inspector to give each instance the same instance name that you specified in the Component inspector.

Add or remove Drag objects and Target objects

To change the default number of four objects and four targets, add or delete objects and targets. A Drag and Drop learning interaction can contain up to eight Drag objects and up to eight Target objects.

See also

"Adding, naming, and registering assets" on page 334

"Name and register graphic distractors" on page 335

Add a Drag object or Target object

1 Create a movie clip symbol containing the graphics for the object. For example, to add a seventh interaction to an interaction that has six types of fruit, create a graphic of the seventh fruit and place it in the library.

2 Select the Drag and Drop learning interaction in the Timeline, and then drag the symbol from the Library panel to the Stage.

3 In the Property inspector, name the instance.

4 Add the instance name to the Component inspector for the Drag and Drop object.

Remove a Drag and Drop object

1 Select the Drag and Drop instance to remove, and delete it from the Stage.

2 Select the Drag and Drop component (to the left of the Stage in the quiz template), and display the Component inspector by opening it from the Property inspector.

3 Remove the deleted object's instance name from the appropriate column in the Component inspector.

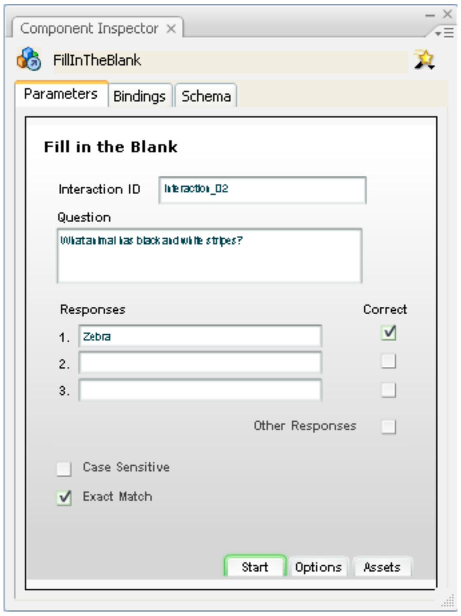
Configure a Fill In The Blank interaction

The Fill In The Blank interaction uses a question text field, a user entry text field, a control button, and a feedback text field.

1 If you are not using a quiz template, place the learning interaction on the Stage. If you are using a quiz template, select the frame on the Interactions layer that contains the Fill In The Blank interaction (Frame 3, if you did not add or remove keyframes).

2 Break the movie clip apart (Modify > Break Apart), display the Component inspector, and type the interaction ID and the question.

3 In the Component inspector, do one of the following to enter up to three possible correct answers:



Fill in the Blank

Interaction ID: Interaction_02

Question: What animal has black and white stripes?

Responses	Correct
1. Zebra	☑
2.	□
3.	□
Other Responses	□

□ Case Sensitive

☑ Exact Match

- Type the text for the responses that the user can enter that are considered correct responses. Select Correct to the right of the correct responses.
- To set up the interaction to accept all responses except those you type, enter the invalid responses in the list and deselect Correct to the right of them. Select Other Responses to indicate that all other responses are correct.

4 Specify whether the matching responses are valid only if they match the case of the text you entered (by selecting Case Sensitive) or if they are valid regardless of the capitalization the user enters (by deselecting Case Sensitive).

5 Specify whether the matching response must be an exact match. If you select Exact Match, a correct response matches only if the user enters the text exactly as it appears in your response. With Exact Match deselected, an answer is considered correct if it contains the correct word. For example, if the answer is zebra and the user enters `striped zebra`, the answer is considered correct. This feature does not work if the correct answer is more than one word.

See also

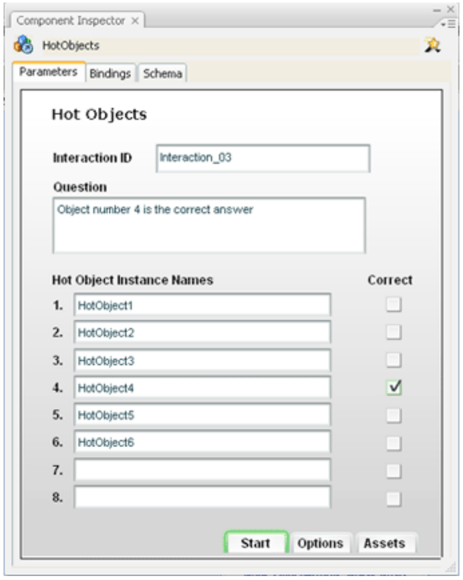
"Configure a Learning Interaction component" on page 329

Configure a Hot Object interaction

The Hot Object interaction accepts up to eight hot objects. The default sample uses six hot objects.

1 If you are not using a quiz template, place the learning interaction on the Stage. If you are using a quiz template, select the frame on the Interactions layer that contains the Hot Object interaction (Frame 5, if you did not add or remove keyframes).

2 Break the movie clip apart (Modify > Break Apart), display the Component inspector, and type the interaction ID and the question.



Hot Objects

Interaction ID: Interaction_03

Question: Object number 4 is the correct answer

	Hot Object Instance Names	Correct
1.	HotObject1	☐
2.	HotObject2	☐
3.	HotObject3	☐
4.	HotObject4	☑
5.	HotObject5	☐
6.	HotObject6	☐
7.		☐
8.		☐

3 For each object, select or deselect Correct to specify whether the object is considered a correct or incorrect response when the user clicks it. Multiple correct selections are allowed.

4 Select each instance of the Hot Object interaction on the Stage (you can delete the placeholder instances and place your own movie clip instances on the Stage). Use the Property inspector to give each instance the same instance name that you specified in the Component inspector.

See also

"Configure a Learning Interaction component" on page 329

Add or remove hot object distractors

To change the default number of six distractors (choices), add more distractors or delete existing distractors. A Hot Object learning interaction can contain up to eight hot object distractors.

See also

"Name and register graphic distractors" on page 335

Add a hot object distractor

1 Create a movie clip symbol containing the graphics for the hot object distractor. For example, to add a seventh choice to an interaction that has six types of fruit, create a graphic of the seventh fruit and place it in the library.

2 Select the Hot Object component on the Stage, and then drag the symbol from the Library panel to the Stage.

3 In the Property inspector, name the instance.

4 Add the instance name to the Component inspector for the hot object.

The component does the rest of the work automatically at runtime.

Remove a hot object distractor

1 Select the Hot Object movie clip instance to remove, and delete it from the Stage.

2 Select the Hot Object component (to the left of the Stage in the quiz template) and the Component inspector by opening it from the Property inspector.

3 Remove the deleted object's instance name from the list in the Component inspector.

Configure a Hot Spot interaction

The Hot Spot learning interaction sets up an interaction in which the user responds by clicking an object (or objects) onscreen.

1 If you are not using a quiz template, place the learning interaction on the Stage. If you are using a quiz template, select the frame on the Interactions layer that contains the Hot Spot interaction (Frame 5, if you did not add or remove keyframes).

2 Break the movie clip apart (Modify > Break Apart), display the Component inspector, and type the interaction ID and the question.

3 For each Hot Spot interaction, select or deselect Correct to specify whether the object is considered a correct or incorrect response when the user clicks it. Multiple correct answers are allowed.

4 To give each movie clip the same instance name that you specified in the Component inspector, delete the placeholder instances on the Stage. Place your movie clips on the Stage and use the Property inspector.

See also

"Configure a Learning Interaction component" on page 329

Add or remove hot spot distractors

A Hot Spot learning interaction can contain up to eight distractors (choices). To change the default number of six distractors, add more distractors or delete existing distractors.

In general, you place the hot spot distractors over another graphic that the user is really intended to see. To make hot spot assets semi-invisible during authoring to visualize this effect, turn the alpha effect setting down on each hot spot. The interaction scripts override this setting at runtime.

See also

"Name and register graphic distractors" on page 335

Add a hot spot distractor

1 Create a movie clip symbol containing the graphics for the distractor object. For example, to add a seventh choice to an image that will have six hot spots, create a movie clip of the seventh graphic and place it in the library.

2 Select the hot spot component on the Stage and drag the symbol from the Library panel to the Stage.

3 In the Property inspector, name the instance.

4 Add the instance name to the Component inspector for the hot spot.

Remove a hot spot distractor

1 Select the hot spot instance to remove and delete it from the Stage.

2 Select the Hot Spot component (to the left of the Stage in the quiz template) and display the Component inspector (Window > Component Inspector).

3 Remove the deleted object's instance name from the list in the Component inspector.

Configure a Multiple Choice interaction

In a Multiple Choice interaction, the user responds to a question with multiple answers. Multiple correct answers are allowed.

1 If you are not using a quiz template, place the learning interaction on the Stage. If you are using a quiz template, select the frame on the Interactions layer that contains the Multiple Choice interaction (Frame 6, if you did not add or remove keyframes).

2 Break the movie clip apart (Modify > Break Apart), display the Component inspector and type the interaction ID and the question.

3 Type the possible responses for the interaction (A to E).

***Note:** You do not need to provide five responses. You can delete a response, but replace it or move any following responses to the previous box, if necessary, so that no blank boxes are between responses.*

4 To specify whether each response is considered correct or incorrect, select or deselect Correct. Multiple correct answers are allowed.

See also

"Configure a Learning Interaction component" on page 329

Add or remove multiple-choice distractors

A Multiple Choice learning interaction can contain up to eight distractors (choices). Change the default number of six distractors by adding more distractors or by deleting existing distractors.

See also

"Name and register graphic distractors" on page 335

Add a multiple-choice distractor

1 Select the frame with the Multiple Choice learning interaction in the Timeline.

2 Open the Flash UI Components folder in the Library panel (Window > Library) and drag a CheckBox component to the Stage.

3 In the Property inspector, name the instance.

4 Add the instance name to the Component inspector for the multiple-choice distractor.

Remove a multiple-choice distractor

1 Select the CheckBox instance to remove, and delete it from the Stage.

2 Select the Multiple Choice component (to the left of the Stage in the quiz template) and display the Component inspector (Window > Component Inspector).

3 Remove the deleted object's instance name from the list in the Component inspector.

Configure a True or False interaction

The True or False interaction includes a question text field, two RadioButton components, a control button, and a feedback text field. There are no other distractor options to configure.

1 If you are not using a quiz template, place the learning interaction on the Stage. If you are using a quiz template, select the frame on the Interactions layer that contains the True or False interaction (Frame 7, if you did not add or remove keyframes).

2 Break the movie clip apart (Modify > Break Apart), display the Component inspector, and type the interaction ID and the question.

3 In the Question box, type the text of the question to ask the user.

4 Select Correct to specify which answer, True or False, is the correct response for the interaction. Change these responses to Correct or Incorrect by changing the text of the distractors. For example, type `A. Correct` and `B. Incorrect` in the Distractors boxes.

See also

"Configure a Learning Interaction component" on page 329

Set feedback options for a learning interaction

Feedback options control the text that the user sees before and while responding to an interaction.

1 Select the interaction component (to the left of the Stage in the quiz template).

2 If the Component inspector is not already visible, open it from the Property inspector, and click Options.

3 Select Feedback for the interaction to present comments to users before and after they submit a response. Enter a comment for the following:

- For Tries, enter the number of times that a user is given to provide a correct response.
- For Initial Feedback, enter the feedback that appears before the user has interacted with the quiz—for example, `Click an object and drag it to the matching object.`
- For Correct Feedback, enter the feedback that appears if the user's response is correct—for example, `Yes, that is correct.`
- For Incorrect Feedback, enter the feedback that appears if the user's response is incorrect and tries is set to 1—for example, `No, that is incorrect.`
- For Additional Tries, enter the feedback that appears if the user's response is incorrect and tries is set to more than 1—for example, `No, that is incorrect. Try again.`

Note: *Because users are allowed one try only for the True or False learning interaction, no Additional Tries field exists for that interaction.*

Set Knowledge Track options for a learning interaction

Knowledge Track is an automatic data-tracking feature that lets you transmit student performance data to an LMS or to other back-end tracking systems. Knowledge Track works with both AICC- and SCORM-compliant learning management systems. Knowledge Track captures or stores student information internal to the Flash application and transmits that data to an HTML page.

To successfully send data to a tracking system, embed the SWF file containing your learning interactions in an HTML page and select the HTML template in Publish Settings for either Flash with AICC Tracking or Flash with SCORM Tracking. To support an AICC-compliant LMS, the HTML that embeds the SWF file needs to be part of a frameset.

The tracking data captured and that Knowledge Track transmits is based on an industry standard for courseware-to-tracking-system communications, the AICC specification version 2. This standard specifies the following data elements for each interaction.

Set values for these data elements by using the Component inspector for an interaction:

- InteractionID
- ObjectiveID
- Weighting

Other data elements are automatically set or calculated:

- Question Type
- Correct Response
- User Response
- Result
- Date/Time
- Latency

1 Select the Learning Interaction component to the left of the Stage in the quiz template.

2 Open the Component inspector from the Property inspector and click Options.

3 Select Knowledge Track if you are using the learning interaction in a document created by using a quiz template and for the learning interaction to send data to a server-side learning management database.

4 (Optional) If the interaction is related to an objective that is set up in the LMS, enter that name in the Objective ID box to specify an objective for the interaction. Tracking still works if you leave the Objective ID box blank.

5 Specify the Weighting value for the interaction. The quiz templates use this parameter to calculate the score in the Results page. The default value is 1. Weighting indicates the relative importance of a question. Enter any numeric value. If all learning interactions have a weight of 1, they are all scored equally. A weight of 2 counts twice as much as a weight of 1 and half as much as a weight of 4. For example, give advanced questions a weight of 3 and beginning-level questions a weight of 1.

See also

“Prepare Flash learning interactions for web hosting” on page 348

Set navigation options for a learning interaction

1 Select the Learning Interaction component to the left of the Stage in the quiz template.

2 If the Component inspector is not already visible, open it from the Property inspector, and then click Options at the bottom of the panel.

3 Under Navigation, specify how the interaction proceeds after the user submits a response for this interaction:

- Select Off to disable navigation if you are using the quiz templates, because the templates include their own navigation.
- Select Next Button to require that the user click Next after submitting a response. In the GoTo Action field, select either Stop or Play. The Next button is a Button component that you can use with stand-alone interactions independent of the quiz template.

To navigate to a labeled frame instead of the next frame, enter a frame label in the GoTo Label box.

The default text for the Next button is Next Question.

- Select Auto GoTo Next Frame to have the interaction proceed to the next frame after the user submits a response.

If Feedback is deselected and Knowledge Track is selected, enable the Auto GoTo Next Frame feature. This feature submits a score after evaluation and immediately navigates to the next frame for the next interaction.

Note: *If Feedback is selected or Knowledge Track is deselected, Auto GoTo Next Frame is reset to Next Button and an error message appears in the Output panel.*

Set control button labels for a learning interaction

All interactions use an instance of the same control buttons: Check Answer, Submit, Next Question, and Reset. The only exception is the True Or False interaction, which does not use a Reset button. Change the label for the instance of each button by using the Component inspector.

1 Select the Learning Interaction component to the left of the Stage in the quiz template.

2 Open the Component inspector from the Property inspector and click Assets at the bottom of the panel.

3 Edit the label name under Control Button Labels.

4 To view the new labels on the buttons, select Control > Test Movie.

Changing the appearance of a learning interaction

Changing the appearance of assets

After you add a learning interaction to the Stage and break it apart, you can place and size most assets. However, making changes to certain Flash components, such as buttons, check boxes, and radio buttons in learning interactions, requires less common processes.

Change the images in a graphical learning interaction

For Drag and Drop, Hot Spot, and Hot Object learning interactions, you change the appearance of the graphic *distractors* (the selectable choices) in the interaction to suit the purposes of your course.

1 If it is not broken apart, select the learning interaction movie clip and select Modify > Break Apart.

2 Select the placeholder graphical objects and delete them.

3 To add your own custom Drag objects, create or import a graphic and convert it to a movie clip symbol (Modify > Convert to Symbol).

4 Place an instance of the symbol in the desired location on the Stage. In the Property inspector, type the name of the movie clip instance, such as DragA, in the Instance Name box.

5 In the Component inspector for the interaction, enter the same instance name (such as DragA) of the movie clip in the appropriate Name box. The Component inspector should include only the unique instance names of the movie clips that you're using for the current interaction.

6 Repeat steps 3 to 5 for additional graphical objects in the interaction.

***Note:** Use Flash user interface components to create the graphics for navigation buttons and for True Or False and Multiple Choice interactions. Only intermediate and advanced users should change these graphics. You can also resize and slightly modify the appearance of these graphics. For more information, see "Customizing Components" in Using Components.*

Changing buttons, check boxes, and radio buttons

The learning interactions use the Flash user interface (UI) Button, CheckBox, RadioButton and TextInput components. You must use these UI components in the learning interaction movie clips. The learning interaction scripts use the internal features of the UI components to function properly.

The quiz templates already contain all the necessary UI components for each interaction. To use UI components in Flash MX or later documents, publish the SWF file using ActionScript 2.0.

Sizing

To scale the Button components used for the Control button, Reset button, and CheckBox, RadioButton, and TextInput components, select the component and change its settings in the Property inspector.

UI component graphics

There is a defined process for changing the skin of a component. For more information, see "Editing component skins in a document" in *Using Components*.

UI component text

To change the text characteristics of a UI component, use the `GlobalStyleSheet` object. See "Customizing Components" in *Using Components*.

See also

"Set control button labels for a learning interaction" on page 345

Using components in a learning interaction

To use Flash UI components with a learning interaction, you add the UI components to the interaction assets and name their instances. You then register the instance names with the component associated with that interaction. Each learning interaction already contains the appropriate UI components as named instances.

For complete documentation on the UI components, see *Using Components*.

***Note:** UI components have an associated Component inspector. The learning interaction scripts override the UI Component inspector at runtime. You do not need to fill out individual parameters for each Button, CheckBox, RadioButton or TextInput component.*

See also

"Adding, naming, and registering assets" on page 334

Tracking to AICC- or SCORM-compliant learning management systems

AICC- and SCORM-compliant learning management systems

The Flash learning interactions and quiz templates allow easy communication with both AICC- and SCORM-compliant LMSs. The code built into both the Flash documents and the corresponding HTML/JavaScript files send properly formatted data to the LMS. The stand-alone interactions send question data, while the quiz templates track the score and overall time spent.

Because of differences between the two tracking standards (AICC and SCORM), the files created with the Flash learning interactions and the quiz templates have compliance differences.

To be SCORM-compliant, content must call an initialize command when it is first started, or before any other tracking commands are sent to the LMS. The Flash with SCORM HTML template was designed to initialize communication with a SCORM-compliant LMS when the file is loaded. It also sends a finish communication to the LMS when the file is unloaded, if the finish command wasn't explicitly sent previously.

Files created by using both the Flash learning interactions and the quiz templates can send tracking data to an AICC- and SCORM-compliant LMS. Individual interactions do not send overall score and tracking data, but they can send interaction or question data.

Files created by using the quiz templates to comply with either AICC or SCORM standards do not read data from the LMS into the Flash file.

AICC communication overview

When a student takes an AICC-compliant quiz, the following events occur:

- The LMS is opened.
- The student logs in to the LMS.
- The student navigates through the course structure to find an assignable unit (AU). In this case, assume it's a Flash quiz, built using a Flash quiz template.
- The student starts the Flash content (the quiz).
- The content is located on a web server. To track properly, the Flash file must be embedded in the Flash AICC tracking frameset.
- The LMS creates two parameters that are appended to the end of the URL: AICC_URL and AICC_SID. When the content is launched, the final URL looks something like the following:

http://myserver/flashcontent.htm?AICC_URL=http://mylmsserver/trackingurl.asp&AICC_SID=12345

- The student progresses through the quiz.
- The Flash learning interaction sends the tracking data to the LMS through the HTML/JavaScript tracking files. The tracking data is sent when the student answers a question or progresses to the next page.

***Note:** Communication with the LMS, and data tracking, is not exposed to the user.*

SCORM communication overview

When a student takes a SCORM-compliant quiz, the following events occur:

- The LMS is initialized.
- The student logs in to the LMS.
- The student starts a quiz built using a Flash quiz template.
- The content is embedded in the Flash/SCORM HTML template, which is opened in a SCORM-compliant frameset.

Note: *This frameset is not exposed to the user.*

The LMS creates the SCORM-compliant frameset, which includes all the necessary functions to communicate back to the LMS.

- The student progresses through the quiz.
- The Flash file sends the tracking data to the LMS through the HTML/JavaScript tracking files.

Prepare Flash learning interactions for web hosting

For web users to see your Flash application, embed it in a web page.

Prepare an AICC-compliant learning interaction for web hosting

1 Open the document in Flash.

2 Select File > Publish Settings.

3 In the Publish Settings dialog box, check that (at least) both Flash (SWF) and HTML files are selected in the Formats panel.

4 Click the HTML tab at the top of the Publish Settings dialog box, and select the Flash With AICC Tracking template from the Template pop-up menu.

5 Click Publish and close the dialog box.

6 Place the files that publishing the Flash file produces and any linked files (such as mp3 or FLV) on the web server in the same directory. Additional files are created if Detect Flash Version is selected in the HTML tab of the Publish Settings dialog box. Copy all the HTML files to your web server, but not the FLA file.

7 Open the Learning Extensions Srvr Files folder, which is located in the Flash 8 program folder in the en/First Run/HTML/Learning Extensions folder. Copy the contents of this folder (frameset.htm, results.htm, and the scripts folder) to the same web server directory as the SWF file and the HTML file published in Flash.

8 Open the new copy of the frameset.htm file in a text editor.

The following lines are found in the frameset.htm file:

```
<frameset frameborder="0" border="0" framespacing="0" rows="*,1">0
<frame src="Untitled-1.htm" name="content" frameborder="0">
<frame src="results.htm" name="cmiresults" scrolling="0" frameborder="0">
```

9 In the second line, change Untitled-1.htm to the name of the HTML file you published in Flash (typically the HTML filename specified in the formats Tab of Publish Settings).

The main file references any HTML files that the publishing process creates. For example, if myQuiz.htm, myQuiz_content.htm, and myQuiz_alternate.htm were created by publishing the document, myQuiz.htm replaces Untitled-1.htm in the frameset.htm file. Then, myQuiz.htm calls myQuiz_content.htm and myQuiz_alternate.htm when necessary.

10 Start the LMS system (or create the AICC Course Descriptor Files) that references the frameset.htm file.

Prepare a SCORM-compliant learning interaction for web hosting

1 Open the document in Flash.

2 Select File > Publish Settings.

3 In the Publish Settings dialog box, check that (at least) both Flash (SWF) and HTML are selected in the Formats panel.

4 Click the HTML tab at the top of the Publish Settings dialog box, and select the Flash With SCORM Tracking template from the Template pop-up menu.

5 Click Publish, and close the dialog box.

6 Place the files that publishing the Flash file produces on the web server in the same directory.

7 Start the LMS system and reference the name of the HTML file. Set the LMS to launch the SCORM tracking frameset.

Extending learning interaction scripts

Accessing cumulative tracking data through the SessionArray

Cumulative tracking data is useful to extend tracking or analysis of the interactions, including creating customized quiz environments and creating quizzes in a format different from that of the Flash quiz templates. Data is tracked through the SessionArray as follows:

- When the Flash application is run, the first interaction component to load creates an Array on the level of the interaction assets.
- The component then creates an instance of the LToolBox global class in index0 of the Array. The instance of `LToolBox` is a storage place for all of the interaction's data. Data is set or retrieved from the instance by using predefined property names.
- When the Timeline moves to the second interaction, that interaction's component creates an instance of `LToolBox global class` in index1 of the SessionArray.
- When the Timeline moves to the third interaction, that interaction's component creates an instance of `LToolBox global class` in index2 of the SessionArray. It continues with index3, index4, and so on, until all interactions are in an index.
- At the end of a series of interactions, all the data processed during those interactions are available and organized.

Note: *SessionArray and session are reserved keywords on the level where the interactions reside. Do not use these words as identifiers for other data. The SessionArray is used the same way in the stand-alone interactions and the quiz interactions.*

Tracking properties available in the SessionArray

The property names reference standard interaction tracking values for both AICC and SCORM LMSs. Retrieve an interaction's properties by referencing its location in the following command:

```
SessionArray[n].[property_name]
```

For example, to reference the `interaction_id` value for interaction #1, use the following command:

```
SessionArray[0].interaction_id
```

To reference the result value for interaction #2, use the following command:

```
SessionArray[1].result
```

Predefined property names

Property name	Description
interaction_id	Unique interaction name
interaction_type	Type of interaction
objective_id	Objective identification number
weighting	Weighting value for this interaction instance; some interactions can have more weight than others
correct_response	Formatted correct response returned from the user parameters
student_response	Formatted student response returned from the evaluation
result	Result of the evaluation
latency	Elapsed time during this interaction session
dateStamp	Date when the interaction occurs
timeStamp	Time when the interaction starts

All the methods and properties of the `LToolBox` global class are available in each SessionArray index.

Basic structure of the Learning Interaction scripts and components

The Learning Interaction components are the center of the e-learning setup. They collect user parameters and build SessionArray and interaction event handling functions on the level of the interaction assets. That is, they accept user parameters and configure the environment and assets accordingly. To examine how these components work, open the scripts in the Library panel.

Most of the scripts reside in one of two places. The first is the `LToolBoxglobalclass` script. This script processes data storage and data formatting for the interaction. The second script location is in each interaction component. These scripts initialize event handling functions that the interaction assets trigger. This is where the user parameters and interaction assets are initialized and the interaction evaluations scripts reside. Although these scripts are built on the component level, they are initialized on the same level as the interaction assets and submit data to the SessionArray on the interaction assets level.

To explore the scripts or add to them, in the library panel, select the 1_GlobalClass folder to access the `LGlobalClass` movie clip that contains the `LToolBoxglobalclass` script . Look in the 2_Components folders to access each Learning Interaction component script. Each script is split into commented sections that are described at the top of the script. Most of the script sections are built in functions for modularity.

Review or edit the LToolboxClass script

The `LToolboxClass` script creates a built-in object that each interaction can use for data storage and basic functionality. The data pattern and functionality shared by all interactions is defined in this script. Access the `LToolboxClass` script from the library.

1. In the Library panel, select Learning Interactions > Assets > Controls > ComponentSuperClass.
2. In the ComponentSuperClass folder, double-click the SuperClass movie clip to open it in symbol-editing mode.
3. In the Timeline for the movie clip, select Frame 1 and open the Actions panel (Window > Actions).
4. Review or edit the script.

Chapter 17: Creating accessible content

You can create content that is accessible to all users, including those with disabilities, using the accessibility features that Adobe® Flash® CS3 Professional provides in the authoring environment user interface, taking advantage of ActionScript™ designed to implement accessibility. As you design accessible Flash applications, consider how users might interact with the content and follow recommended design and development practices.

For a tutorial about accessible content, see Create Accessible Flash Content on the Flash Tutorials page at www.adobe.com/go/learn_fl_tutorials.

For a sample of accessible Flash content, see the Flash Samples page at www.adobe.com/go/learn_fl_samples. Download and decompress the Samples zip file and navigate to the Accessibility\AccessibleApplications folder to access the sample.

About accessible content

Worldwide accessibility standards

Many countries have adopted accessibility standards based on the standards developed by the World Wide Web Consortium (W3C). The W3C publishes the *Web Content Accessibility Guidelines*, a document that prioritizes actions designers should take to make web content accessible. For information about the Web Accessibility Initiative, see the W3C website at w3.org.

In the United States, the law that governs accessibility is commonly known as Section 508, which is an amendment to the U.S. Rehabilitation Act.

For additional information about Section 508, see the following websites:

- The US government-sponsored website at section508.gov
- The Adobe accessibility site at www.adobe.com/accessibility/

Adobe Flash Accessibility web page

For the latest information on creating and viewing accessible Flash content, including supported platforms, screen reader compatibility, articles, and accessible examples, see the Flash Accessibility web page at www.adobe.com/go/flash_accessibility/.

Understanding screen reader technology

Screen readers are software applications that visually impaired users can use to navigate a website and read the web content aloud. To enable a screen reader to read nontextual objects in your application, such as vector art and animations, use the Accessibility panel to associate a name and description with the object. The keyboard shortcuts you define can allow users to use the screen reader to navigate through your document with ease.

To expose graphic objects, use the Accessibility panel or ActionScript to provide a description.

You cannot control how any screen reader behaves; you can control only the content, which you can mark up in your Flash applications to expose the text and ensure that screen reader users can activate the controls. You decide which objects in the Flash application are exposed to screen readers, provide descriptions for them, and decide the order in which they are exposed to screen readers. You cannot force screen readers to read specific text at specific times or control the manner in which that content is read. Test your applications with a variety of screen readers to ensure that they perform as you expect.

See also

“Create and name a keyboard shortcut” on page 361

“Using Flash to enter accessibility information for screen readers” on page 355

Platform requirements

You can only create Flash content designed for use with screen readers with Windows platforms. Viewers of Flash content must have Macromedia Flash® Player 6 from Adobe or later and Internet Explorer on Windows 98 or later.

Flash and Microsoft Active Accessibility (Windows only)

Flash Player is optimized for Microsoft Active Accessibility (MSAA), which provides a descriptive and standardized way for applications and screen readers to communicate. MSAA is available only for Windows operating systems. For more information on Microsoft Accessibility Technology, visit the Microsoft Accessibility website at www.microsoft.com/enable/default.aspx.

The Windows ActiveX (Internet Explorer plug-in) version of Flash Player 6 supports MSAA, but Windows Netscape and Windows stand-alone players do not.

***Important:** MSAA is currently not supported in the opaque windowless and transparent windowless modes. (These modes are options in the HTML Publish Settings panel, available for use with the Windows version of Internet Explorer 4.0 or later, with the Flash ActiveX control.) To make your Flash content accessible to screen readers, avoid using these modes.*

Flash Player makes information about the following types of accessibility objects available to screen readers that use MSAA.

Dynamic or static text The principal property of a text object is its name. To comply with MSAA conventions, the name is equal to the contents of the text string. A text object can also have an associated description string. Flash uses the static or dynamic text immediately above or to the left of an input text field as a label for that field.

***Note:** Any text that is a label is not passed to a screen reader, but is used as the name of the object that it labels. Labels are never assigned to buttons or text fields that have author-supplied names.*

Input text fields Have a value, an optional name, a description string, and a keyboard shortcut string. An input text object’s name can come from a text object that is above or to the left of it.

Buttons Have a state (pressed or not pressed), support a programmatic default action that causes the button to depress momentarily, and optionally have a name, a description string, and a keyboard-shortcut string. Flash uses any text entirely inside a button as a label for that button.

***Note:** For accessibility purposes, Flash Player considers movie clips used as buttons with button event handlers such as* `onPress` *to be buttons, not movie clips.*

Components Provide special accessibility implementation.

Movie clips Exposed to screen readers as graphic objects when they do not contain any other accessible objects, or when you use the Accessibility panel to provide a name or a description for a movie clip. When a movie clip contains other accessible objects, the clip itself is ignored, and the objects inside it are made available to screen readers.

***Note:** All Flash Video objects are treated as simple movie clips.*

See also

"Using Flash to enter accessibility information for screen readers" on page 355

"Using accessible components" on page 365

"Creating accessibility with ActionScript" on page 362

Basic accessibility support in Flash Player

By default, the following objects are defined as accessible in all Flash documents and are included in the information that Flash Player provides to screen reader software. This generic support for documents that do not use any accessibility features includes the following:

Dynamic or static text Text is transferred to the screen reader program as a name, but with no description.

Input text fields Text is transferred to the screen reader. No names are transferred, except where a labeling relationship is found for the input text, such as a static text field positioned close to the input text field. No descriptions or keyboard shortcut strings are transferred.

Buttons The state of the button is transferred to the screen reader. No names are transferred, except where labeling relationships are found, and no descriptions or keyboard shortcut strings are transferred.

Documents The document state is transferred to the screen reader, but with no name or description.

Accessibility for hearing-impaired users

Include captions for audio content that is integral to comprehending the material. A video of a speech, for example, might require captions for accessibility, but a quick sound associated with a button probably wouldn't.

Methods to add captions to a Flash document include the following:

- Add text as captions, ensuring that the captions are synchronized with the audio in the Timeline.
- Use Hi-Caption Viewer, a component available from Hi Software that works with Hi-Caption SE for use with Flash (see www.adobe.com/go/accessible_captions). *Captioning Macromedia Flash Movies with Hi-Caption SE*, a white paper, explains how to use Hi-Caption SE and Flash together to create a captioned document (see www.adobe.com/go/accessibility_papers).

Provide animation accessibility for the visually impaired

You can change the property of an accessible object during SWF file playback. For example, to indicate changes that take place on a keyframe in an animation. However, different vendor's screen readers treat new objects on frames differently. Some screen readers might read only the new object, whereas other screen readers might re-read the entire document.

To reduce the chance of causing a screen reader to emit extra "chatter" that can annoy users, avoid animating the text, buttons, and input text fields in your document. Also, avoid making your content loop.

Flash Player can't determine the actual text content of features such as Text Break Apart to animate text. Screen readers can only provide accurate accessibility to information-carrying graphics such as icons and gestural animation, if you provide names and descriptions for these objects in your document or for the entire Flash application. You can also add supplementary text to your document or shift important content from graphics to text.

1 Select the object for which you want to change the accessibility properties.

2 Select Window > Other Panels > Accessibility.

3 Change the properties for the object.

Alternatively, use ActionScript to update accessibility properties.

See also

"Make an entire Flash application accessible" on page 359

"Creating accessibility with ActionScript" on page 362

Testing accessible content

When you test your accessible Flash applications, follow these recommendations:

- Download several screen readers and test your application by playing it in a browser with the screen reader enabled. Check that the screen reader is not attempting to "talk over" places in your document where you inserted separate audio. Several screen reader applications provide a demonstration version of the software as a free download; test as many screen readers as you can to ensure compatibility across screen readers.
- Test interactive content and verify that users can navigate your content effectively using only the keyboard. Different screen readers work in different ways when processing input from the keyboard; your Flash content might not receive keystrokes as you intended. Test all keyboard shortcuts.

Using Flash to enter accessibility information for screen readers

Flash for screen readers and accessibility

Screen readers read aloud a description of the content, read text, and assist users as they navigate through the user interfaces of traditional applications such as menus, toolbars, dialog boxes, and input text fields.

By default, the following objects are defined as accessible in all Flash documents and are included in the information that Flash Player provides to screen reader software:

- Dynamic text
- Input text fields
- Buttons
- Movie clips
- Entire Flash applications

Flash Player automatically provides names for static and dynamic text objects, which are the contents of the text. For each of these accessible objects, you can set descriptive properties for screen readers to read aloud. You can also control how Flash Player decides which objects to expose to screen readers—for example, you can specify that certain accessible objects are not exposed to screen readers at all.

The Flash Accessibility panel

The Flash Accessibility panel (Window > Other Panels > Accessibility) lets you provide accessibility information to screen readers and set accessibility options for individual Flash objects or entire Flash applications.

Note: *Alternatively, use ActionScript code to enter accessibility information.*

If you select an object on the Stage, you can make that object accessible and specify options and tab order for the object. For movie clips, you can specify whether child object information is passed to the screen reader (the default when you make an object accessible).

With no objects selected on the Stage, use the Accessibility panel to assign accessibility options for an entire Flash application. You can make the entire application accessible, make child objects accessible, have Flash label objects automatically, and give specific names and descriptions to objects.

All objects in Flash documents must have instance names for you to apply accessibility options to them. Create instance names for objects in the Property inspector. The instance name is used to refer to the object in ActionScript.

The following options are available in the Accessibility panel:

Make Object Accessible (Default) Instructs Flash Player to pass the accessibility information for an object to a screen reader. When the option is disabled, accessibility information for the object is not passed to screen readers. Disabling this option as you test content for accessibility can be useful because some objects might be extraneous or decorative and making them accessible could produce confusing results in the Screen Reader. You can then apply a name manually to the labeled object, and hide the labeling text by deselecting Make Object Accessible. When Make Object Accessible is disabled, all other controls on the Accessibility panel are disabled.

Make Child Objects Accessible (Movie clips only; Default) Instructs Flash Player to pass child object information to the screen reader. Disabling this option for a movie clip causes that movie clip to appear as a simple clip in the accessible object tree, even if the clip contains text, buttons, and other objects. All objects in the movie clip are then hidden from the object tree. This option is useful mainly for hiding extraneous objects from screen readers.

Note: *If a movie clip is used as a button—it has a button event handler assigned to it, such as* `onPress` *or* `onRelease`*— the Make Child Objects Accessible option is ignored because buttons are always treated as simple clips, and their children are never examined, except in the case of labels.*

Auto Label Instructs Flash to automatically label objects on the Stage with the text associated with them.

Name Specifies the object name. Screen readers identify objects by reading these names aloud. When accessible objects don't have specified names, a screen reader might read a generic word, such as *Button*, which can be confusing.

Important: *Do not confuse object names specified in the Accessibility panel with instance names specified in the Property inspector. Giving an object a name in the Accessibility panel does not give it an instance name.*

Description Lets you enter a description of the object to the screen reader. The screen reader reads this description.

Shortcut Describes keyboard shortcuts to the user. The screen reader reads the text in this text field. Entering keyboard shortcut text here does not create a keyboard shortcut for the selected object. You must provide ActionScript keyboard handlers to create shortcut keys.

Tab Index (Adobe® Flash® CS3 Professional only) Creates a tab order in which objects are accessed when the user presses the tab key. The tab index feature works for keyboard navigation through a page, but not for screen reader reading order.

For more information, see the Flash Accessibility web page at www.adobe.com/go/flash_accessibility/.

For a tutorial about accessible content, see Create Accessible Flash Content on the Flash Tutorials page at www.adobe.com/go/learn_fl_tutorials.

For a sample of accessible Flash content, see the Flash Samples page at www.adobe.com/go/learn_fl_samples. Download and decompress the Samples zip file and navigate to the Accessibility\AccessibleApplications folder to access the sample.

See also

"Creating accessibility with ActionScript" on page 362

"Create and name a keyboard shortcut" on page 361

"Turn off automatic labeling for an object and specify a name" on page 360

Selecting names for buttons, text fields, and entire Flash applications

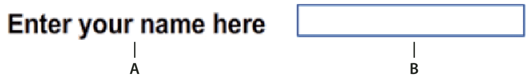
Use the Accessibility panel in the following ways to assign names to buttons and input text fields so that the screen reader identifies them appropriately:

- Use the auto label feature to assign text adjacent or in the object as a label.
- Enter a specific label in the Accessibility panel name field.

Flash automatically applies the name that you place on top of, in, or near a button or text field as a text label. Labels for buttons must appear within the bounding shape of the button. For the button in the following example, most screen readers would first read the word *button*, then read the text label *Home*. The user can press Return or Enter to activate the button.

A form might include an input text field where users enter their names. A static text field, with the text *Name* appears next to the input text field. When Flash Player discovers such an arrangement, it assumes that the static text object serves as a label for the input text field.

For example, when the following part of a form is encountered, a screen reader reads "Enter your name here."



A. *Static text* ***B.*** *Input text field*

In the Accessibility panel, turn off automatic labeling if it is not appropriate for your document. You can also turn off automatic labeling for specific objects in your document.

See also

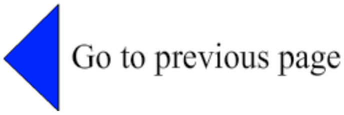
"Turn off automatic labeling for an object and specify a name" on page 360

Provide a name for an object

You can turn off automatic labeling for part of an application and provide names for the objects in the Accessibility panel. If you have automatic labeling turned on, you can select specific objects and provide names for the objects in the Name text field in the Accessibility panel so that the name is used instead of the object text label.

When a button or input text field doesn’t have a text label, or when the label is in a location that Flash Player can’t detect, you can specify a name for the button or text field. You can also specify a name if the text label is near a button or text field, but you don’t want that text to be used as that object’s name.

In the following example, the text that describes the button appears outside and to the right of the button. In this location, Flash Player does not detect the text, and the screen reader does not read it.



To correct this situation, open the Accessibility panel, select the button, and enter the name and description. To prevent repetition, make the text object inaccessible.

***Note:** An object’s accessibility name is unrelated to the ActionScript instance name or ActionScript variable name associated with the object. (This information generally applies to all objects.) For information on how ActionScript handles instance names and variable names in text fields, see About text field instance and variable names in Learning ActionScript 2.0 in Adobe Flash.*

Specify a name and description for a button, text field, or entire Flash application

1 Do one of the following:

- To provide a name for a button or text field, select the object on the Stage.
- To provide a name for an entire Flash application, deselect all objects on the Stage.

2 Select Window > Other Panels > Accessibility.

3 Select either Make Object Accessible (for buttons or text fields) or the default, Make Movie Accessible (for entire Flash applications).

4 Enter a name and description for the button, text field, or Flash application.

Define accessibility for a selected object in a Flash application

1 Select Window > Other Panels > Accessibility.

2 Do one of the following:

- Select Make Object Accessible (the default setting) to expose the object to screen readers and to enable other options in the panel.
- Deselect Make Object Accessible to hide the object from screen readers and disable the other options in the panel.

3 Enter a name and a description for the selected object as needed:

Dynamic text To provide a description for static text, you must convert it to dynamic text.

Input text fields or buttons Enter a keyboard shortcut.

Movie clips Select Make Child Objects Accessible to expose the objects inside the movie clip to screen readers.

***Note:** If you can describe your application in a simple phrase that a screen reader can easily convey, turn off Make Children Accessible, and type a suitable description.*

Make an entire Flash application accessible

After a Flash document is complete and ready to be published or exported, make the entire Flash application accessible.

1 Deselect all elements in the document.

2 Select Window > Other Panels > Accessibility.

3 Select Make Movie Accessible (the default setting) to expose the document to screen readers.

4 Select or deselect Make Children Accessible to expose or omit any accessible objects in the document to screen readers.

5 If you selected Make Movie Accessible in step 2, enter a name and description for the document as needed.

6 Select Auto Label (the default setting) to use text objects as automatic labels for accessible buttons or input text fields contained in the document. Deselect this option to turn off automatic labeling and expose text objects to screen readers as text objects.

Viewing and creating tab order and reading order

The two aspects to tab indexing order are the *tab order* in which a user navigates through the web content and the order in which things are read by the screen reader, called the *reading order*.

Flash Player uses a tab index order from left to right and top to bottom. Customize both the tab and reading order by using the `tabIndex` property in ActionScript (in ActionScript, the `tabIndex` property is synonymous with the reading order).

***Note:** Flash Player no longer requires that you add all of the objects in a FLA file to a list of tab index values. Even if you do not specify a tab index for all objects, a screen reader reads each object correctly.*

Tab order The order in which objects receive input focus when users press the Tab key. Use ActionScript to create the tab order, or if you have Adobe® Flash® CS3 Professional, use the Accessibility panel. The tab index that you assign in the Accessibility panel does not necessarily control the reading order.

Reading order The order in which a screen reader reads information about the object. To create a reading order, use ActionScript to assign a tab index to every instance. Create a tab-order index for every accessible object, not just the focusable objects. For example, dynamic text must have tab indexes, even though a user cannot tab to dynamic text. If you do not create a tab index for every accessible object in a given frame, Flash Player ignores all tab indexes for that frame whenever a screen reader is present, and uses the default tab ordering instead.

See also

"Use ActionScript to create a tab order for accessible objects" on page 364

Create a tab-order index for keyboard navigation in the Accessibility panel

You can create a custom tab-order index in the Accessibility panel for keyboard navigation for the following objects:

- Dynamic text
- Input text
- Buttons
- Movie clips, including compiled movie clips
- Components

- Screens

***Note:** You can also use ActionScript code to create a tab-order index for keyboard navigation.*

Tab focus occurs in numerical order, starting from the lowest index number. After tab focus reaches the highest tab index, focus returns to the lowest index number.

When you move tab-indexed objects that are user-defined in your document, or to another document, Flash retains the index attributes. Check for and resolve index conflicts (for example, two different objects on the Stage with the same tab-index number).

***Important:** If two or more objects have the same tab index in any given frame, Flash follows the order in which the objects were placed on the Stage.*

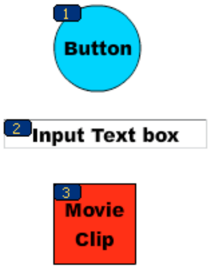
Create and view a tab-order index

1 Select the object in which to assign a tab order.

2 Select Window > Other Panels > Accessibility.

3 If you're providing an index for the selected object only, in the Tab Index text field, enter a positive integer (up to 65535) that reflects the order in which the selected object should receive focus.

4 To view a tab order, select View > Show Tab Order. Tab index numbers for individual objects appear in the upper-left corner of the object.



Tab index numbers

***Note:** Tab indexes created with ActionScript code do not appear on the Stage when the Show Tab Order option is enabled.*

Using sound with screen readers

Sound is the most important medium for most screen reader users. Consider how any sound in your document interacts with the text spoken aloud by screen readers. It might be difficult for screen reader users to hear what their screen readers are saying if your Flash application contains loud sounds.

Specifying advanced accessibility options for screen readers

Turn off automatic labeling for an object and specify a name

1 On the Stage, select the button or input text field for which you want to control labeling.

2 Select Window > Other Panels > Accessibility.

3 Select Make Object Accessible (the default setting).

4 Enter a name for the object. The name is read as the label for the button or text field.

5 To turn off accessibility for the automatic label (and hide it from screen readers), select the text object on the Stage.

6 If the text object is static text, convert it to dynamic text (in the Property inspector, select Text type > Dynamic Text).

7 Deselect Make Object Accessible.

Hide an object from the screen reader

You can hide a selected object from screen readers, and you can decide to hide accessible objects that are contained inside a movie clip or Flash application and expose only the movie clip or Flash application to screen readers.

Note: *Only hide objects that are repetitive or convey no content.*

When an object is hidden, the screen reader ignores the object.

1 On the Stage, select the button or input text field to hide from the screen reader.

2 Select Window > Other Panels > Accessibility.

3 In the Accessibility panel, do one of the following:

- If the object is a movie clip, button, text field, or another object, deselect Make Object Accessible.
- If the object is the child of a movie clip, deselect Make Child Objects Accessible.

Create and name a keyboard shortcut

You can create a keyboard shortcut for an object, such as a button, so users can navigate to it without listening to the contents of an entire page. For example, you can create a keyboard shortcut to a menu, a toolbar, the next page, or a submit button.

To create a keyboard shortcut, write ActionScript code for an object. If you provide a keyboard shortcut for an input text field or button, you must also use the ActionScript Key class to detect the key the user presses during Flash content playback. See Key in the *ActionScript 2.0 Language Reference*. See Capturing keypresses in *Learning ActionScript 2.0 in Adobe Flash*.

Select the object and add the name of the keyboard shortcut to the Accessibility panel so the screen reader can read it.

Test your Flash content with multiple screen readers. Keyboard shortcut functionality also depends on the screen reader software used. The key combination Control+F, for example, is a reserved keystroke for both the browser and the screen reader. The screen reader reserves the arrow keys. Generally, you can use the 0 to 9 keys on the keyboard for keyboard shortcuts, however, screen readers increasingly use even these keys.

See also

“Testing accessible content” on page 355

Create a keyboard shortcut

1 On the Stage, select the button or input text field to create a keyboard shortcut for.

2 Select Window > Other Panels > Accessibility.

3 In the Shortcut field, type the name of the keyboard shortcut, using the following conventions:

- Spell out key names, such as Control or Alt.
- Use capital letters for alphabetic characters.
- Use a plus sign (+) between key names, with no spaces (for example, Control+A).

Important: *Flash does not check that the ActionScript to code the keyboard shortcut was created.*

Map Control+7 to myButton instance

1 Select the object on the Stage, display the Accessibility panel, and in the Shortcut field, type **`Control+7`**.

2 Enter the following ActionScript 2.0 code in the Actions panel:

```
function myOnPress() {
    trace( "hello" );
}
function myOnKeyDown() {
    if (Key.isDown(Key.CONTROL) && Key.getCode() == 55) // 55 is key code for 7
    {
        Selection.setFocus(myButton);
        myButton.onPress();
    }
}
var myListener = new Object();
myListener.onKeyDown = myOnKeyDown;
Key.addListener(myListener);
myButton.onPress = myOnPress;
myButton._accProps.shortcut = "Ctrl+7"
Accessibility.updateProperties();
```

Note: *The example assigns the Control+7 keyboard shortcut to a button with an instance name of myButton and makes information about the shortcut available to screen readers. In this example, when you press Control+7, the* `myOnPress` *function displays the text "hello" in the Output panel. See addListener (IME.addListener method) in ActionScript 2.0 Language Reference.*

Creating accessibility with ActionScript

About ActionScript and accessibility

You can create accessible documents with ActionScript code. For accessibility properties that apply to the entire document, you can create or modify a global variable called `_accProps`. See the `_accProps` property in *ActionScript 2.0 Language Reference*.

For properties that apply to a specific object, you can use the syntax `instancename._accProps`. The value of `_accProps` is an object that can include any of the following properties:

Property	Type	Equivalent selection in the Accessibility panel	Applies to
`.silent`	Boolean	Make Movie Accessible/Make Object Accessible (inverse logic)	Entire documents Buttons Movie clips Dynamic text Input text
`.forceSimple`	Boolean	Make Child Objects Accessible (inverse logic)	Entire documents Movie clips
`.name`	string	Name	Entire documents Buttons Movie clips Input text
`.description`	string	Description	Entire documents Buttons Movie clips Dynamic text Input text
`.shortcut`	string	Shortcut	Buttons Movie clips Input text

***Note:** With inverse logic, a value of `true` in ActionScript corresponds to a check box that is not selected in the Accessibility panel, and a value of `false` in ActionScript corresponds to a selected check box in the Accessibility panel.*

Modifying the `_accProps` variable has no effect by itself. You must also use the `Accessibility.updateProperties` method to inform screen reader users of Flash content changes. Calling the method causes Flash Player to re-examine all accessibility properties, update property descriptions for the screen reader, and, if necessary, send events to the screen reader that indicate changes have occurred.

When updating accessibility properties of multiple objects at once, include only a single call to `Accessiblity.updateProperties` (too frequent updates to the screen reader can cause some screen readers to become too verbose).

See the Accessibility.updateProperties method in the *ActionScript 2.0 Language Reference.*

Implementing screen reader detection with the Accessibility.isActive() method

To create Flash content that behaves in a specific way if a screen reader is active, use the `Accessibility.isActive()` ActionScript method, which returns a value of `true` if a screen reader is present, and `false` otherwise. You can then design your Flash content to perform so that it's compatible with screen reader use (for example, by hiding child elements from the screen reader). For more information, see isActive (Accessibility.isActive method) in *ActionScript 2.0 Language Reference.*

For example, you could use the `Accessibility.isActive()` method to decide whether to include unsolicited animation. Unsolicited animation happens without the screen reader doing anything, which can be confusing for screen readers.

The `Accessibility.isActive()` method provides asynchronous communication between the Flash content and Flash Player; a slight real-time delay can occur between the time the method is called and the time when Flash Player becomes active, returning an incorrect value of `false`. To ensure that the method is called correctly, do one of the following:

- Instead of using the `Accessibility.isActive()` method when the Flash content first plays, call the method whenever you need to make a decision about accessibility.
- Introduce a short delay of one or two seconds at the beginning of your document to give the Flash content enough time to contact Flash Player.

For example, you can use an `onFocus` event to attach this method to a button. This approach generally gives the SWF file enough time to load and you can assume a screen reader user will tab to the first button or object on the Stage.

Use ActionScript to create a tab order for accessible objects

To create the tab order with ActionScript code, assign the `tabIndex` property to the following objects:

- Dynamic text
- Input text
- Buttons
- Movie clips, including compiled movie clips
- Timeline frames
- Screens

Provide a complete tab order for all accessible objects. If you create a tab order for a frame and you don’t specify a tab order for an accessible object in the frame, Flash Player ignores all the custom tab-order assignments. Additionally, all objects assigned to a tab order, except frames, must have an instance name specified in the Instance Name text field of the Property inspector. Even items that are not tab stops, such as text, need to be included in the tab order if they are to be read in that order.

Because static text cannot be assigned an instance name, it cannot be included in the list of the `tabIndex` property values. As a result, a single instance of static text anywhere in the SWF file causes the reading order to revert to the default.

To specify a tab order, assign an order number to the `tabIndex` property, as the following example shows:

```
_this.myOption1.btn.tabIndex = 1
_this.myOption2.txt.tabIndex = 2
```

See `tabIndex` in `Button`, `MovieClip`, and `TextField` in the *ActionScript 2.0 Language Reference.*

You can also use the `tabChildren()` or `tabEnabled()` methods to assign custom tab order. See `MovieClip.tabChildren`, `MovieClip.tabEnabled`, and `TextField.tabEnabled` in the *ActionScript 2.0 Language Reference.*

See also

“Create a tab-order index for keyboard navigation in the Accessibility panel” on page 359

Using accessible components

A core set of UI components accelerates building accessible applications. These components automate many of the most common accessibility practices related to labeling, keyboard access, and testing and help ensure a consistent user experience across rich applications. Flash includes the following set of accessible components:

- SimpleButton
- CheckBox
- RadioButton
- Label
- TextInput
- TextArea
- ComboBox
- ListBox
- Window
- Alert
- DataGrid

Accessible Flash components must contain ActionScript that defines their accessible behavior. For information on which accessible components work with screen readers, see the Flash Accessibility web page at www.adobe.com/go/flash_accessibility/.

For general information about components, see "About Components" in *Using ActionScript 2.0 Components*.

For each accessible component, enable the accessible portion of the component with the `enableAccessibility()` command. This command includes the accessibility object with the component as the document is compiled. Because no simple way exists to remove an object after it is added to the component, these options are disabled by default. Therefore, it's important that you enable accessibility for each component. Perform this step only once for each component; you do not need to enable accessibility for each instance of a component for a given document. See "Button component", "CheckBox component", "ComboBox component", "Label component", "List component", "RadioButton component", and "Window component" in the *ActionScript 2.0 Components Language Reference*.

Chapter 18: Working with screens

The Screens feature in Adobe® Flash® CS3 Professional does not support ActionScript™ 3.0. To use screens, you must start with an ActionScript 2.0-based FLA file.

Screen-based documents and the screen authoring environment

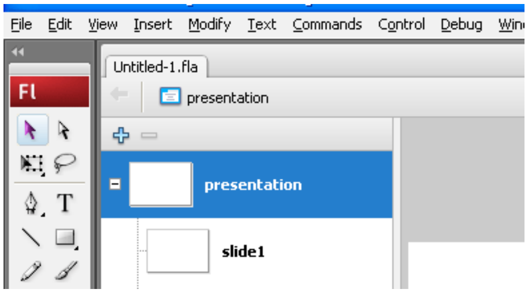
About the screen authoring environment

Screens provide an authoring user interface with structural building blocks that make it easy for you to create complex, hierarchical Flash documents, such as slide presentations or form-based applications.

Note: *The Screens feature in Flash does not support ActionScript 3.0. To use screens, you must start with an ActionScript 2.0-based FLA file.*

Screens let you structure complex applications without using multiple frames and layers in the Timeline, and without viewing the Timeline.

When you author a screen-based document, the screens are arranged in a structured hierarchy that you create. To structure the document, nest screens in a branching tree. You can easily preview and modify the structure of a screen-based document.



Detail of default pasteboard for a new Flash Slide Presentation. Screen thumbnails appear in the Screen Outline pane on the left side of the pasteboard, and the Timeline is collapsed.

To author a screen-based document, you first create a Slide Presentation or Form Application document. Then you add screens, configure the screens and add content, and add behaviors to create controls and transitions for the screens.

Slide presentations and screens, and form applications and screens

There are two types of screen-based documents. The type of document you select determines the type of default screen in the document.

- A Flash Slide Presentation uses the slide screen as the default screen type. A slide screen is designed for a sequential presentation.

- A Flash Form Application uses the form screen as the default screen type. A form screen is designed for a nonlinear, form-based application.

Although each document has a default screen type, you can include and mix slide screens and form screens in any screen-based document.

To control screens, set parameters for slide or form screens in the Property inspector or use ActionScript. For more information, see “Screen class”, “Form class”, and “Slide class”, in the *ActionScript 2.0 Components Language Reference*.

Slide screens let you create Flash documents with sequential content, such as a slide show. The default runtime behavior lets users navigate sequentially through slide screens, using the left and right arrow keys. Sequential screens can overlay one another so that the previous screen remains visible when the next slide is viewed. Screens can continue playing after they are hidden. To automatically manage the visibility of each screen, use slide screens.

Form screens let you create structured form-based applications, such as online registration or e-commerce forms. By default, to create the navigation structure with form screens, you must write ActionScript code. To manage the visibility of individual screens yourself, use form screens.

To learn more about creating screen-based documents that use ActionScript, see “About organizing code for screens in” in *Using ActionScript 2.0 Components*.

See also

“Creating controls and transitions for screens with behaviors” on page 376

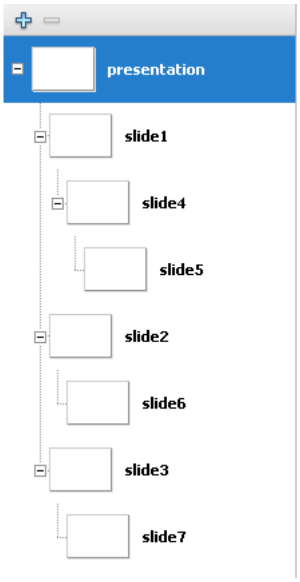
Document structure and hierarchy

Each document has a master screen at the top level. In a Flash Slide Presentation, the top-level screen is called Presentation by default. In a Flash Form Application, the top-level screen is called Application by default.

The top-level screen is the container for everything that you add to the document, including other screens. You can place content on the top-level screen. You cannot delete or move the top-level screen.

Screens are similar to nested movie clips in some ways: Child screens inherit the behavior of their parents, and you use target paths in ActionScript to send messages from one screen to another. However, screens do not appear in the library, and you cannot create multiple instances of a screen.

You can add multiple screens to a document, and you can nest screens within other screens, in as many levels as you want. A screen that is inside another screen is the *child* of that screen. A screen that contains another screen is the *parent* of that screen. If a screen is nested several layers deep, all the screens above that screen are its *ancestors*. Screens that are at the same level are *sibling* screens. All screens nested in another screen are its *descendants*. A child screen contains all the content of its ancestor screens.



Top-level slide has three children, Slide 1, Slide 2, and Slide 3. Slide 1 has one child and one grandchild

To learn more about creating screen-based documents that use ActionScript, see "About organizing code for screens in" in *Using ActionScript 2.0 Components*.

Using preloaders with screen-based documents

To include a preloader (a separate SWF file that loads your primary SWF) with your screen-based document, create the preloader as a separate SWF file (non-screen-based), and load the SWF file for the screen-based document from within the preloader SWF.

You cannot create a preloader within a screen-based document, because all screens in a document are located on the first frame of the root Timeline, so you cannot call or load other frames.

Working with screens

Add screens to a document

You can add a default-type screen or select a screen type at the same level as the currently selected screen. The new screen is a *sibling* screen of the selected screen. You can also add a nested screen one level below the currently selected screen. To view all screens in a document, use the Screen Outline pane.

When you add screens to a document, Flash exhibits the following default behaviors:

- Uses the screen type of the document (slide type for a Slide Presentation or form type for a Form Application) for the new screen. To select to insert a screen of another type, use the Insert Screen Type command in the Screens context menu.
- Inserts the first screen you add directly after the top-level screen, one level below it.
- Inserts a screen after the currently selected screen, at the same level. If the document contains nested screens below the currently selected screen, the new screen is added after the nested screens, at the same level as the selected screen.

- Flash inserts a nested screen directly after the currently selected screen, and nested one level down. If the document contains a nested screen or screens below the currently selected screen, the new screen is added after all nested screens already in place, one level below the selected screen.

Add a default-type screen or a screen of a specified type at the current screen level

1 Select a screen in the Screen Outline pane.

2 To add a default-type screen, do one of the following:

- Press Enter or Return.
- Click the Insert Screen (+) button at the top of the Screen Outline pane.
- Select Insert > Screen.
- Select Insert Screen from the Screens context menu.

3 To add a screen of a specified type, select Insert Screen Type from the context menu and select a screen type.

Add a nested screen of the default type

1 Select a screen in the Screen Outline pane.

2 Do one of the following:

- Press Enter or Return.
- Select Insert > Nested Screen.
- Select Insert Nested Screen from the Screens context menu.

Use the Screen Outline pane

The Screen Outline pane at the left of the Document window displays thumbnails of each screen in the current document, in a collapsible tree view. The tree represents the structural hierarchy of the document. Nested screens are indented below the screen that contains them.

When you add a screen to a document, the screen appears in the Screen Outline pane.

To hide and show nested screens, collapse and expand the tree. You can hide, show, and resize the Screen Outline pane.

- To display the screen on the Stage, click on a screen thumbnail in the Screen Outline pane.
- To hide or show the Screen Outline pane, select Window > Other Panels > Screens.
- To expand or collapse the tree, click the Plus (+) or Minus (-) button next to a screen to show or hide the screens nested within it (Windows), or click the triangle next to a screen to show or hide the screens nested within it (Macintosh).
- To resize the Screen Outline pane, drag the dividing line between the Screen Outline pane and the Document window.

See also

“Creating controls and transitions for screens with behaviors” on page 376

View the context menu for a screen

The Screens context menu contains commands for working with screens.

❖ Right-click (Windows) or Control-click (Macintosh) a screen thumbnail in the Screen Outline pane.

Using timelines with screens

Each screen in a screen-based Flash document has its own Timeline that is collapsed by default. To work with frames or layers, open the Timeline (Window > Timeline).

You cannot view or modify the root Timeline of a screen-based Flash document.

You can add frames, keyframes, and layers, and manipulate content on a screen's Timeline.

In the Timeline, nested screens work much as nested movie clips do, with some exceptions.

See also

"How screens interact with ActionScript" on page 378

Using the Movie Explorer with screens

To view and organize the contents of a document containing screens, use the Movie Explorer. The Movie Explorer handles documents that contain screens much as it handles documents that do not contain screens, with the following exceptions:

- The Movie Explorer shows only the contents of the current screen (the screen selected in the Screens Outline pane).
- You cannot view scenes in the Movie Explorer because a document with screens cannot contain scenes.

Select, move, and edit screens

When you select an individual screen in the Screen Outline pane, the screen appears in the Document window. To apply modifications to several screens at once, select multiple contiguous or discontiguous screens in the Screen Outline pane. When you select multiple screens, the contents of the first screen selected appear in the Screen Outline pane.

By default, the contents of a slide screen are not visible when you show the screen's parent in the Document window (the Hide Screen context menu option is selected). To show the contents of a slide screen when its parent appears, deselect this option. When the Hide Screen context menu option is deselected, you can select the child slide screen on the Stage. This feature affects display during authoring only, not runtime playback. The Hide Screen context menu option is deselected for form screens by default. To hide child form screens in the display during authoring, turn the option on.

You can cut, copy, paste, and drag screens in the Screen Outline pane to change their position in the document, and you can remove screens from a document.

***Note:** The terms child, parent, and ancestor refer to the hierarchical relationships of nested screens.*

See also

"Document structure and hierarchy" on page 367

"Speed up document display" on page 485

View a screen in the Document window

❖ Do any of the following:

- To view that screen, click a screen thumbnail in the Screen Outline pane.
- To navigate to the screen, use the keyboard keys with the Screen Outline pane in focus.

- To navigate through the screens, select View > Go To and select the screen name from the submenu, or select First, Previous, Next, or Last.
- Click the Edit Screen button at the right side of the edit bar and select the screen name.

Select multiple screens in the Screen Outline pane

- To select multiple contiguous screens, Shift-click the first and the last screen to select.
- To select multiple discontiguous screens, Control-click (Windows) or Command-click (Macintosh) each screen.

Edit an item on a screen

❖ Select the item in the Document window.

View the contents of a child screen when the parent screen appears

❖ To turn off the Hide feature (Hide Screen is selected for slide screens by default.), click Hide Screen in the child screen's context menu.

Select a child screen on the Stage

1 Deselect Hide Screen.

2 Select the parent screen in the Screen Outline pane.

3 Click in the contents of the child screen on the Stage.

Edit an item on an ancestor screen of the current screen

❖ Double-click the item in the Document window.

Note: *By default, items on ancestor screens of the current screen are dimmed in the Document window.*

Fully render all items on ancestor screens

❖ Select View > Preview Mode > Full.

Cut or copy a screen

❖ Do any of the following:

- Right-click (Windows) or Control-click (Macintosh) the screen, and select Cut or Copy from the context menu.
- Select Edit > Cut or Edit > Copy.

Paste a screen

❖ Do either of the following:

- After cutting or copying the screen, right-click (Windows) or Control-click (Macintosh) another screen and select Paste from the context menu. The cut or copied screen is pasted after the selected screen. To nest the pasted screen within the selected screen, select Paste Nested Screen from the context menu.
- After cutting or copying the screen, select Edit > Paste In Center or Edit > Paste In Place.

Drag a screen in the Screen Outline pane

❖ Using the mouse, drag the screen to any other position in the Screen Outline pane. Release the mouse button when the screen is in the desired position. To nest a screen within another screen, drag it towards the right side of the Screen Outline pane below the intended parent.

Remove a screen

❖ Do one of the following:

- Right-click (Windows) or Control-click (Macintosh) the screen, and select Cut or Delete from the context menu.
- Select the screen, and click the Delete Screen (-) button at the top of the Screen Outline pane.
- Press Backspace (Windows) or Delete (Macintosh).

About screen names

By default, screens are named with their default type, in the order in which they are created: slide1, slide2, form1, form2, and so on. The creation order does not necessarily reflect the order of the screens in the Screen Outline pane. For example, you could create three sibling screens, slide1, slide2, and slide3. If you then create a nested screen directly below slide1, the nested screen is slide4.

To rename a screen, including the top-level screen, double-click the screen name in the Screen Outline pane and enter the new name. Screen names must be unique in a document. For example, a document can have only one screen named *Quiz_Page*.

The default screen name is used as the instance name, which is used in ActionScript to control a screen. If you change the default screen name, the instance name is updated with the new name; likewise, if you change the instance name, the screen name is updated. The linkage identifier for the screen is also identical to the screen name, and it is updated when the screen name or instance is updated.

Instance names must conform to the following requirements:

- The name must not contain any spaces.
- The first character must be a letter, underscore (_), or dollar sign ($).
- Each subsequent character must be a letter, number, underscore, or dollar sign.
- The instance name must be unique.

You can also change the instance name in the Property inspector.

To learn more about creating screen-based documents that use ActionScript, see “About organizing code for screens in” in *Using ActionScript 2.0 Components*.

See also

“Set properties and parameters for a screen” on page 373

Undoing and redoing commands with screens

To undo and redo the following actions performed on screens, use the Edit > Undo and Edit > Redo menu commands: adding, cutting, copying, pasting, deleting, and hiding a screen. The following actions performed on screens are recorded in the History panel: adding a screen, adding a nested screen, selecting a screen, renaming a screen, and deleting a screen.

See also

“Undo, Redo, and Repeat commands” on page 47

Use Find And Replace in a document with screens

You can search for a text string, font, color, symbol, sound file, video file, or imported bitmap file.

You can search for elements in the entire document or in the current screen.

1 Select Edit > Find And Replace.

2 Do one of the following:

- To search the entire document, select Current Document from the Search In pop-up menu.
- To search a screen, click in the Screen Outline pane, and select Current Screen from the Search In pop-up menu.

See also

"Find and Replace" on page 76

Accessibility in the Flash screens authoring environment

Using keyboard shortcuts rather than the mouse, users can navigate a document and use interface elements, including screens, panels, the Property inspector, dialog boxes, the Stage, and objects on the Stage.

Accessibility support for screen-based documents is similar to support for other documents, except that when keyboard shortcuts are used to navigate panels (Control+F6 in Windows or Command+F6 in Mac OS), the Screen Outline pane receives focus the first time the keyboard shortcut is used. For other documents, the Timeline receives focus first).

The Screen Outline pane receives focus only the first time you cycle through the panels. That is, if you arrive at the last panel and press the keyboard shortcut again, the Screen Outline pane is skipped, and the next panel receives focus.

To cycle through individual screens in the Screen Outline pane, you use the arrow keys.

See also

"Creating accessible content" on page 352

Adding content to screens

Adding media content to screens

Add media content to the screen that is currently selected in the Screen Outline pane.

Set properties and parameters for a screen

To set properties and parameters for individual screens, use the Property inspector. On the left side of the Property inspector, you can view the instance name, width, height, and *x* and *y* coordinates of a screen.

- The instance name is a unique name assigned to a screen, used when you target the screen in ActionScript. Each screen is assigned a default instance name, based on its default name in the Screen Outline pane. The instance name and default screen name are also identical to the linkage identifier for the screen. If you update the instance name, the default screen name and the linkage identifier are also updated.

- Width and height are specified in pixels. The values in the W and H fields are read-only. Width and height are determined by the screen contents. To make sure the registration point stays in the same relative position when the screen width and height change, use the Auto Snap option.
- The *x* and *y* coordinates of a screen are specified in pixels. Move a child screen on the Stage by changing its *x* and *y* coordinates. Change the registration point of a screen using the registration grid.

To control screen behavior during playback, set parameters for slide and form screens.

Change the instance name of a screen

1 Select a screen in the Screen Outline pane.

2 Select Window > Properties > Properties.

3 On the left side of the Property inspector, enter a name in the Instance Name box.

***Note:** If you update the instance name, the screen name in the Screen Outline pane and the linkage identifier for the screen also update.*

Move a child screen on the Stage

1 Deselect Hide Screen for the child screen.

2 Select the screen's parent in the Screen Outline pane, and select the child screen's content on the Stage.

3 Select Window > Properties > Properties.

4 In the Property inspector, enter new values for the *x* and *y* coordinates, drag the child screen to another location on the Stage, or use the Align panel.

Specify the ActionScript class and registration point of a screen

Specify the ActionScript class of the screen and its registration point on the Properties tab of the Property inspector:

- The ActionScript class specifies what class the screen belongs to. The class determines what methods and properties are available for the screen. By default, slide screens are assigned to the `mx.screens.Slide` class, and form screens are assigned to the `mx.screens.Form` class. You can assign the screen to a different class.
- The registration grid indicates the position of the screen registration point in relation to its content. By default, the registration point of a slide screen is in the center and Auto Snap is on. By default, the registration point of a form screen is in the upper left corner and Auto Snap is off. To change the registration point, use the grid. To keep the registration point in the same position in relation to screen contents, even when you add, remove, or reposition the screen contents, use the Auto Snap option.

The height and width of a screen are determined by its content. Therefore, the center of a screen cannot be the center of the Stage.

***Note:** If you changed the coordinate grid setting in the Info panel in another Flash document, the coordinate grid for the screen registration point can reflect that change. To check the Info panel coordinate grid setting, open a non-screen-based Flash document or select something on the Stage that is not a screen, and select Window > Info. To change settings in the Info panel while working in a screen-based document, deselect all screens before you open the panel.*

See also

"Get information about instances on the Stage" on page 215

Change the ActionScript class of a screen

1 Select a screen in the Screen Outline pane.

2 Select Window > Properties > Properties.

3 In the Property inspector, click the Properties tab.

4 Enter a class name in the Class Name box. For more information on ActionScript classes, see Classes in *Learning ActionScript 2.0 in Adobe Flash*.

Change the registration point of a screen

1 Select a screen in the Screen Outline pane.

2 Select Window > Properties > Properties.

3 Click the Properties tab and click a point in the registration grid.

Clicking a different registration point in the registration grid changes the point on the screen that is used as the registration point. When this is done, the registration point moves in relation to the screen content, but the screen itself does not move.

Setting parameters for a screen

To control how the screen appears and behaves during playback, set parameters on the Parameters tab of the Property inspector. Different parameters are available for slide and form screens.

The following parameters are available only for slide screens:

- The autoKeyNav parameter determines whether the slide uses default keyboard handling to control navigation to the next or previous slide. When `autoKeyNav` is set to `true`, pressing the Right Arrow key or the Spacebar advances to the next slide, and pressing the Left Arrow key moves to the previous slide. When `autoKeyNav` is set to `false`, no default keyboard handling takes place. When `autoKeyNav` is set to `inherit` (the default setting), the slide inherits its `autoKeyNav` setting from its parent. If the slide's parent is also set to `inherit`, the parent's ancestors are examined until one is found with its `autoKeyNav` parameter set to `true` or `false`. If a slide is a root slide, setting `autoKeyNav` to `inherit` yields the same result as setting it to `true`.

***Note:** This property can be set independently for each slide, and it affects keyboard handling when that slide has focus.*

- The overlayChildren parameter specifies whether child screens overlay one another on the parent screen during playback. When `overlayChildren` is set to `true`, child screens overlay one another. For example, suppose you have two children, Child 1 and Child 2, which are bullet points on the parent screen. If the user clicks a Next button and displays Child 1, then clicks Next again and displays Child 2, Child 1 remains visible when Child 2 appears. When `overlayChildren` is set to `false` (the default setting), Child 1 is removed from the display when Child 2 appears. This parameter affects only the immediate children of a slide, not nested descendants.
- The playHidden parameter specifies whether a slide continues to play if it is hidden after being shown. When `playHidden` is set to `true` (the default setting), the slide continues to play when the slide is hidden after being shown. When `playHidden` is set to `false`, the slide stops playing if it is hidden, and resumes playing at Frame 1 if it is shown again.

One parameter is available only for form screens: The visible parameter indicates whether a screen is visible or hidden at runtime. When `visible` is set to `true`, the screen is visible at runtime. When `visible` is set to `false`, the screen is hidden. This property does not affect the visibility of the screen in the authoring environment.

The following parameters are available for slide and form screens:

- The autoload parameter indicates whether the content should load automatically (`true`), or wait to load until the `Loader.load()` method is called (`false`). The default value is `true`. This parameter is inherited from the Loader component.
- The contentPath parameter is an absolute or relative URL indicating the file to load when the `Loader.load()` method is called. A relative path must point to the SWF file loading the content. The URL must be in the same subdomain as the URL where the Flash content currently resides. For use in Flash Player or with the Test Movie command, all SWF files must be stored in the same folder, and the filenames cannot include folder or disk-drive specifications. The default value is undefined until the load starts. This parameter is inherited from the Loader component.

Specify parameter settings for a screen

1 Select a screen in the Screen Outline pane.

2 Select Window > Properties > Properties.

3 In the Property inspector, click the Parameters tab.

4 Click the setting for a parameter, and select a setting from the pop-up menu.

Creating controls and transitions for screens with behaviors

To create controls and transitions for screens, use behaviors. Controls enable the flow between screens—for example, you can go to another screen, hide a screen, or show a screen. Transitions create visual animations that play as the Flash document display changes from one screen to another.

Behaviors are built-in ActionScript scripts that you add to an object, such as a screen, to control that object. Behaviors let you add the power, control, and flexibility of ActionScript coding to your document without having to create the ActionScript code yourself. Behaviors are available for a variety of objects in Flash, including movie clips, text fields, and video and sound files.

To learn more about creating screen-based documents that use ActionScript, see "About organizing code for screens in" in *Using ActionScript 2.0 Components*.

Add controls to screens using behaviors

To add a control to a screen using a behavior, attach the behavior to a trigger—a button, movie clip, or screen—and target the screen to affect with the behavior. Select the event that triggers the behavior.

To control slide screens, add the following behaviors: Go to First Slide, Go to Last Slide, Go to Next Slide, Go to Previous Slide, and Go to Slide (specify slide name).

***Note:** Go to Next Slide and Go to Previous Slide move to screens on the same level, not to parents or children.*

To control slide or form screens, add the following behaviors: Show a Specified Screen (if the screen was previously hidden) or Hide a Specified Screen (if the screen was previously shown).

1 Select the button, movie clip, or screen that you want to trigger the behavior.

2 In the Behaviors panel, click the Add (+) button.

3 Select Screen, and select the desired control behavior from the submenu.

4 If the behavior requires that you select a target screen, the Select Screen dialog box appears. Select the target screen in the tree control. Click Relative to use a relative target path, or Absolute to use an absolute target path, and click OK.

***Note:** Some behaviors select a target screen by default; for example, the Go to First Slide screen automatically targets the first screen. These behaviors do not show the Select Screen dialog box.*

5 In the Event column, click in the row for the new behavior and select an event from the list. This specifies the event that triggers the behavior—for example, a user clicking a button, a movie clip loading, or a screen receiving focus. The list of available events depends on the type of object you use to trigger the behavior.

See also

“Document structure and hierarchy” on page 367

“Using absolute and relative target paths” on page 72

Adding transitions to screens using behaviors

Screen transition behaviors let you add animated transitions between screens. To add a transition using a behavior, you attach the behavior directly to a screen.

You can select the direction of a transition: In plays the animation as the screen first appears in the document; Out plays the animation as the screen disappears from the document. You can also specify the duration in seconds.

Easing options let you modify the transition to achieve different effects.

Some transitions have additional parameters that you can modify. Parameters appear in the Transitions dialog box when you select the transition.

Follow these guidelines when adding transitions:

- For most situations, the In option is recommended.
- Use the In option when applying a transition that uses the `on(reveal)` event.
- Use the Out option when applying a transition that uses the `on(hide)` event.
- Do not add an Out transition immediately before an In transition in a presentation.
- To attach the same transition to all children of a given slide, attach the single transition to the `on(revealChild)` or `on(hideChild)` event of the parent, rather than duplicating the transition on all child slides.

Add a transition behavior

1 Select the screen to apply the behavior to.

2 In the Behaviors panel, click the Add (+) button.

3 Select Screen > Transition from the submenu.

4 Select a transition from the scroll list. An animated preview of the transition plays in the preview window, and a brief description of the transition appears in the description field. The animation changes to reflect options that you select for the transition in the following steps.

5 For Direction, select In to play the transition as the screen appears in the document, and Out to play the transition as the screen disappears from the document.

6 For Duration, enter a time in seconds.

7 For Easing, select an option to define the transition style.

8 If the transition has additional parameters, select options or enter values for those parameters in the fields provided.

9 Click OK.

10 In the Behaviors panel, in the Event column, click in the row for the new behavior and select an event from the list. This action specifies the event that triggers the behavior—for example, the mouse pointer moving over the screen.

Screen instance names, class names, and registration points

The screen name automatically generates the instance name and class name of the screen. You need these identifying labels when you manipulate screens with ActionScript in various ways. To adjust how the screen behaves, change a screen's registration point. You can work with these features in the following ways:

- The instance name is a unique name assigned to a screen, used when you target the screen in ActionScript. Change the instance name in the Property inspector. The instance name is identical to the screen name in the Screen Outline pane and the linkage identifier for the screen. If you update the instance name, the screen name and the linkage identifier also update.

***Note:** Symbol instances, including movie clips, buttons, and graphics, also have instance names.*

- The class name identifies the ActionScript class to which the screen is assigned. By default, a slide screen is assigned to the `mx.screens.Slide` class, and a form screen is assigned to the `mx.screens.Form` class. To modify the methods and properties that are available for the screen, assign the screen to a different class. For more information on ActionScript classes, see Classes in *Learning ActionScript 2.0 in Adobe Flash*.
- The Property inspector indicates the registration point in the *x* and *y* coordinate fields and in the registration grid. You might want to move the registration point for greater control in manipulating screen content. For example, to create a spinning shape in the center of a screen, reposition the screen registration point at the center of the screen and rotate the screen around its registration point.

See also

"Using symbols, instances, and library assets" on page 207

How screens interact with ActionScript

Screens are similar to nested movie clips in the way that they interact with ActionScript. However, some differences exist.

Use the following guidelines for ActionScript with screens:

- When you select a screen in the Screen Outline pane and add ActionScript, the script is added directly to the screen as an object action (much as ActionScript is added directly to a movie clip). Use object actions for simple code (such as creating navigation between screens) and external ActionScript files for more complex code.
- For best results, organize the document structure and finalize screen names before adding ActionScript. If you rename a screen, the instance name is automatically changed, and you must update the instance names in any ActionScript code you write.
- To add a frame action to the Timeline for a screen, select the screen, open the Timeline (Window > Timeline), and select the first frame in the Timeline. Use an external ActionScript file, rather than a frame action, for complex code on a screen.
- You cannot view or manipulate the main Timeline for a screen-based document. However, you can target the main Timeline using `_root` in a target path.

- Each screen is automatically associated with ActionScript, based on its class. You can change the class to which that screen is assigned, and you can set some parameters for a screen in the Property inspector.
- To control screens with ActionScript, use the Screen class, Slide class, and Form class.
- To create interactivity, use components whenever possible. Put no more than 125 total component instances in a single FLA file.
- To create navigation between slides, use `rootSlide`. For example, to get the current slide, use `rootSlide.currentSlide`.
- Do not try to do slide navigation inside of `on(reveal)` or `on(hide)` handlers.
- Do not add an `on(keydown)` or `on(keyup)` event to ActionScript code controlling a screen.

For more information on controlling screens with ActionScript, see “Screen class”, “Form class”, and “Slide class”, in the *ActionScript 2.0 Components Language Reference*.

For information on the Object class and the `onclipEvent()` event handler, see Object and onClipEvent handlerin *ActionScript 2.0 Language Reference*.

To learn more about creating screen-based documents that use ActionScript, see "About organizing code for screens" in *Using ActionScript 2.0 Components*.

See also

“About nested movie clips and parent-child hierarchy” on page 71

Using components with screens

To create complex, structured applications in Flash, use components with screens. Components are especially useful with forms to create structured applications that show data and enable nonlinear user interactivity. For example, use forms to populate a container component.

To create custom navigation between components, use the Focus Manager when you use components with screens. The Focus Manager specifies the order in which components receive focus when a user presses the Tab key to navigate in an application. For example, customize a form application so that a user can press Tab to navigate fields and press Return (Macintosh) or Enter (Windows) to submit the form.

For information on the Focus Manager, see “Creating custom focus navigation” in *Using ActionScript 2.0 Components* and “FocusManager class” in *ActionScript 2.0 Components Language Reference*.

You can also use the Accessibility panel to create a tab order.

See also

“Viewing and creating tab order and reading order” on page 359

Chapter 19: ActionScript

ActionScript™ is the scripting language in Flash. Use ActionScript to make your applications play in a nonlinear way, and to add interesting or complex functionality that cannot be represented in the timeline.

Working with ActionScript

About ActionScript

The ActionScript scripting language lets you add complex interactivity, playback control, and data display to your application. You can add ActionScript in the authoring environment by using the Actions panel, Script window, or an external editor.

ActionScript follows its own rules of syntax, reserved keywords, and lets you use variables to store and retrieve information. ActionScript includes a large library of built-in classes that let you create objects to perform many useful tasks. For more information on ActionScript, see *Programming ActionScript 3.0*, *Learning ActionScript 2.0 in Adobe Flash*, or the *ActionScript Language References*.

You don't need to understand every ActionScript element to begin scripting; if you have a clear goal, you can start building scripts with simple actions.

ActionScript and JavaScript are both rooted in the ECMA-262 standard, the international standard for the ECMAScript scripting language. For this reason, developers who are familiar with JavaScript should find ActionScript immediately familiar. For more information about ECMAScript, go to ecma-international.org.

Using the ActionScript documentation

Because there are multiple versions of ActionScript (2.0 and 3.0), and multiple ways of incorporating it into your FLA files, there are several different ways to learn ActionScript.

This help system describes the graphical user interface for working with ActionScript. This interface includes the Actions panel, Script window, Script Assist mode, Behaviors panel, Output panel, and Compiler Errors panel. These topics apply to all versions of ActionScript.

Other ActionScript documentation from Adobe will help you learn about the individual versions of ActionScript; see *Programming ActionScript 3.0*, *Learning ActionScript 2.0 in Adobe Flash*, *Developing Flash Lite 1.x Applications* or *Developing Flash Lite 2.x Applications*. For information about the ActionScript vocabulary, see the *ActionScript Language Reference* for the version you are working with.

For video tutorials about ActionScript 3.0, the Flash workflow, and components, see the following:

- Getting started with ActionScript 3.0: www.adobe.com/go/vid0129
- Creating interactivity with ActionScript 3.0: www.adobe.com/go/vid0130
- Flash workflow: www.adobe.com/go/vid0132
- Using components: www.adobe.com/go/vid0133

For text tutorials about ActionScript, see www.adobe.com/go/learn_fl_tutorials. The following tutorials are available:

- Create an Application

- Add Interactivity
- Work with Objects and Classes

ActionScript versions

Flash includes more than one version of ActionScript to meet the needs of different kinds of developers and playback hardware.

- ActionScript 3.0 executes extremely fast. This version requires somewhat more familiarity with object-oriented programming concepts than the other ActionScript versions. ActionScript 3.0 is fully compliant with the ECMAScript specification, offers better XML processing, an improved event model, and an improved architecture for working with onscreen elements. FLA files that use ActionScript 3.0 cannot include earlier versions of ActionScript.
- ActionScript 2.0 is simpler to learn than ActionScript 3.0. Although Flash Player runs compiled ActionScript 2.0 code slower than compiled ActionScript 3.0 code, ActionScript 2.0 is still useful for many kinds of projects that are not computationally intensive; for example, more design-oriented content. ActionScript 2.0 is also based on the ECMAScript spec, but is not fully compliant.
- ActionScript 1.0 is the simplest form of ActionScript, and is still used by some versions of the Flash Lite Player. ActionScript 1.0 and 2.0 can coexist in the same FLA file.
- Flash Lite 2.x ActionScript is a subset of ActionScript 2.0 that is supported by Flash Lite 2.x running on mobile phones and devices.
- Flash Lite 1.x ActionScript is a subset of ActionScript 1.0 that is supported by Flash Lite 1.x running on mobile phones and devices.

Ways of working with ActionScript

There are several ways to work with ActionScript.

- Script Assist mode lets you add ActionScript to your FLA file without writing the code yourself. You select actions, and the software presents you with a user-interface for entering the parameters required for each one. You must know a little about what functions to use to accomplish specific tasks, but you don’t have to learn syntax. Many designers and non-programmers use this mode.
- Behaviors also let you add code to your file without writing it yourself. Behaviors are prewritten scripts for common tasks. You can add a behavior and then easily configure it in the Behaviors panel. Behaviors are available only for ActionScript 2.0 and earlier.
- Writing your own ActionScript gives you the greatest flexibility and control over your document, but it requires you to become familiar with the ActionScript language and conventions.
- Components are prebuilt movie clips that help you implement complex functionality. A component can be a simple user interface control, such as a check box, or it can be a complicated control, such as a scroll pane. You can customize a component’s functionality and appearance, and you can download components created by other developers. Most components require you to write some ActionScript code of your own to trigger or control a component. For more information, see About ActionScript 3.0 Components in *Using ActionScript 3.0 Components* or About Components in *Using ActionScript 2.0 Components*.

See also

“Script Assist mode and behaviors” on page 386

Writing ActionScript

When you write ActionScript code in the authoring environment, you use the Actions panel or Script window. The Actions panel and Script window contain a full-featured code editor that includes code hinting and coloring, code formatting, syntax highlighting, syntax checking, debugging, line numbers, word wrapping, and support for Unicode.

- Use the Actions panel to write scripts that are part of your Flash document (that is, scripts that are embedded in the FLA file). The Actions panel provides features such as the Actions toolbox, which gives you quick access to the core ActionScript language elements, and Script Assist mode, in which you are prompted for the elements needed to create scripts.
- Use the Script window if you want to write external scripts—that is, scripts or classes that are stored in external files. (You can also use a text editor to create an external AS file.) The Script window includes code-assistance features such as code hinting and coloring, syntax checking, and auto-formatting.

See also

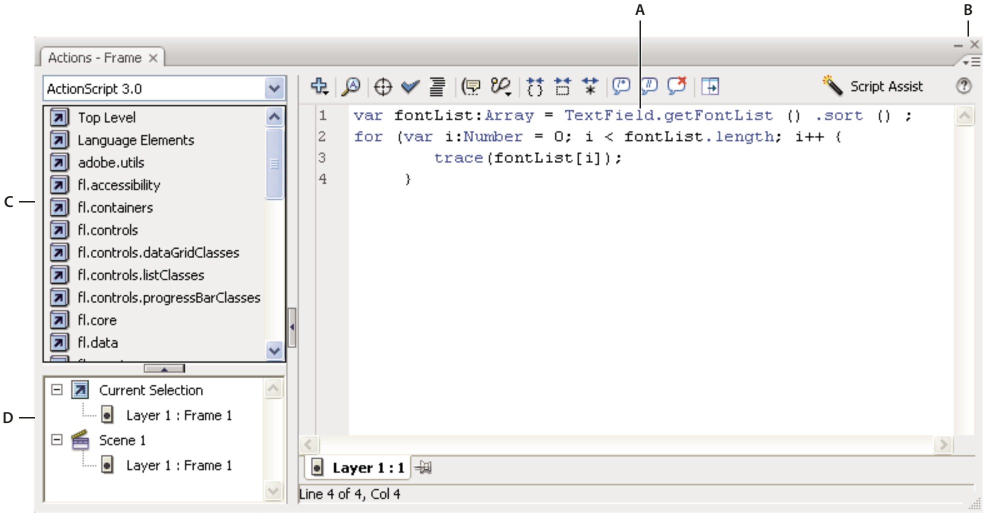
“Debugging ActionScript 1.0 and 2.0” on page 400

“Debugging ActionScript 3.0” on page 410

“Script Assist mode and behaviors” on page 386

Actions panel overview

To create scripts embedded in a FLA file, enter ActionScript directly into the Actions panel. The Actions panel consists of three panes: the Actions toolbox, which groups ActionScript elements by category; the Script navigator, which lets you move quickly between the scripts in your Flash document; and the Script pane, where you type your ActionScript code.



```
var fontList:Array = TextField.getFontList () .sort () ;
for (var i:Number = 0; i < fontList.length; i++ {
        trace(fontList[i]);
      }
```

A. Script pane *B.* Panel menu *C.* Actions toolbox *D.* Script navigator

See also

“Pin scripts in the Actions panel” on page 398

Display the Actions panel

❖ Select Window > Actions or press F9.

Use the Actions toolbox

❖ Insert an ActionScript element into the Script pane by double-clicking or dragging it directly into the Script pane.

The Actions toolbox separates items into categories, and also provides an alphabetical index.

Use the Script pane

❖ Type your code.

Resize the Actions toolbox or Script pane

- Drag the vertical bar that appears between the Actions toolbox and Script pane.
- Click the Show/Hide Toolbox button at the top of the Actions panel.
- Double-click the vertical bar to collapse the Actions toolbox; double-click the bar again to display the Actions toolbox.
- Click the arrow on the vertical bar to expand or collapse the Actions toolbox.

When the Actions toolbox is hidden, you can still use the Add (+) button to access its items.

Use the Script navigator

- Click an item in the Script navigator; the script associated with that item appears in the Script pane, and the playhead moves to that position on the timeline.
- Double-click an item in the Script navigator to *pin* the script (lock it in place).

Print actions

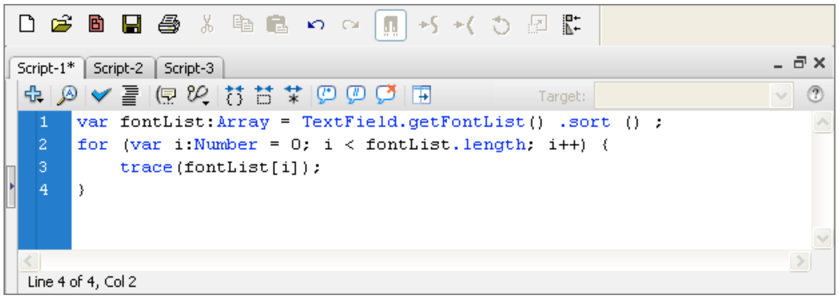
1 From the Actions panel pop-up menu, select Print.

2 Select Options and click Print.

Because the printed copy won’t include information about file it came from, you should include information such as the name of the FLA file in a `comment` action in the script.

Script window overview

The Script window lets you create external script files that you import into your application. These scripts can be ActionScript, Flash Communication, or Flash JavaScript files. The Add (+) menu lists the language elements available for the type of script you create.



```
var fontList:Array = TextField.getFontList() .sort () ;
for (var i:Number = 0; i < fontList.length; i++) {
    trace(fontList[i]);
}
```

Script window

If you have more than one external file open, filenames are displayed on tabs across the top of the Script window.

In the Script window, you can use the following features: the Add (+) menu (which is like the Actions toolbox), find and replace, syntax checking, syntax coloring, auto format, code hinting, code commenting, code collapse, debug options (ActionScript files only), and word wrap. The Script window also lets you display line numbers and hidden characters.

The Script window does not include code-assistance features such as the Script navigator, Script Assist mode, and behaviors. These features are useful only in the context of creating a FLA file, not an external script file.

See also

"Writing and managing scripts" on page 389

Create an external file in the Script window

1 Select File > New.

2 Select the type of external file you want to create (ActionScript file, ActionScript Communication file, or Flash JavaScript file).

Edit an existing file in the Script window

- To open an existing script, select File > Open, and then open an existing AS file.
- To edit a script that is already open, click the document tab that shows the script's name.

Tools in the Actions panel and Script window

The Actions panel and Script window toolbars let you access the code-assistance features that help simplify and streamline coding in ActionScript. The tools are different depending on whether you are using the Actions panel or the Script window.

Add A New Item To The Script Displays the language elements that are also in the Actions toolbox. Select an item to add it to the script.

Find Finds and replaces text in your script.

Insert Target Path (Actions panel only) Helps you set an absolute or relative target path for an action in the script.

Check Syntax Checks for syntax errors in the current script. Syntax errors are listed in the Output panel.

Auto Format Formats your script for proper coding syntax and improved readability. Set autoformatting preferences in the Preferences dialog box, which is available from the Edit menu or from the Actions Panel menu.

Show Code Hint If you've turned off automatic code hinting, use Show Code Hint to display a code hint for the line of code you're working on.

Debug Options (Actions panel only) Sets and removes breakpoints so that you can proceed line by line through your script when debugging. You can use debug options only for ActionScript files, not ActionScript Communication or Flash JavaScript files.

Collapse Between Braces Collapses code that appears between the curly braces or parentheses that currently contain the insertion point.

Collapse Selection Collapses the currently selected code block.

Expand All Expands all collapsed code in the current script.

Apply Block Comment Adds comment markers to the beginning and end of the selected code block.

Apply Line Comment Adds a single-line comment marker at the insertion point, or at the beginning of each line of code in a multiline selection.

Remove Comment Removes comment markers from the current line or all lines of the current selection.

Show/Hide Toolbox Displays or hides the Actions Toolbox.

Script Assist (Actions panel only) In Script Assist mode, you are presented with a user-interface for entering the elements needed to create scripts.

Help Displays reference information for the ActionScript element that is selected in the Script pane. For example, if you click an `import` statement and then click Help, the reference information for `import` appears in the Help panel.

Panel menu (Actions panel only) Contains the commands and preferences that apply to the Actions panel. For example, you can set line numbers and word wrapping, access ActionScript preferences, and import or export scripts.

See also

"Writing and managing scripts" on page 389

"Debugging ActionScript 3.0" on page 410

"Debugging ActionScript 1.0 and 2.0" on page 400

Set ActionScript preferences

Whether you edit code in the Actions panel or the Script window, you can set and modify a single set of preferences.

1 Select Edit > Preferences (Windows) or Flash > Preferences (Macintosh), and then click ActionScript in the Category list.

2 Set any of these preferences:

Automatic Indentation When automatic indentation is turned on, the text you type after an opening parenthesis "(" or opening curly brace "{" is automatically indented according to the Tab Size setting.

Tab Size Specifies the number of characters a new line is indented.

Code Hints Enables code hinting in the Script pane.

Delay Specifies the delay (in seconds) before code hints are displayed.

Font Specifies the font used for your script.

Use Dynamic Font Mapping Checks to ensure that the selected font family can render each character. If not, Flash substitutes a font family that contains the necessary characters.

Open/Import Specifies the character encoding used when you open or import ActionScript files.

Save/Export Specifies the character encoding used when you save or export ActionScript files.

Reload Modified Files Specifies what happens when a script file is modified, moved, or deleted. Select Always, Never, or Prompt.

- **Always** No warning is displayed, and the file is automatically reloaded.
- **Never** No warning is displayed, and the file remains in the current state.

- **Prompt** (Default) A warning is displayed, and you can choose whether to reload the file.

When you build applications with external scripts, this preference helps you avoid overwriting a script that a team member has modified since you opened the application, or avoid publishing the application with older versions of scripts. The warnings let you automatically close a script and reopen the newer, modified version.

Syntax Colors Specifies code coloring in your scripts.

Language Opens the ActionScript Settings dialog boxes, where you can set a classpath for ActionScript 2.0 or 3.0.

See also

"Format code" on page 392

"Use code hints" on page 389

"Import and export scripts" on page 397

"Modify the classpath" on page 414

Script Assist mode and behaviors

About Script Assist mode

If you are new to ActionScript, or if you want to add simple interactivity without having to learn the ActionScript language and its syntax, you can use Script Assist in the Actions panel to help you add ActionScript to your FLA files.

Script Assist lets you build scripts by selecting items from the Actions toolbox. When you click an item once, its description appears at the upper right of the panel. When you double-click an item, it adds the item to the Actions panel Script pane.

In Script Assist mode, you can add, delete, or change the order of statements in the Script pane; enter parameters for actions in boxes above the Script pane; find and replace text; and view script line numbers. You can also *pin* a script—that is, keep a script in the Script pane when you click away from the object or frame.

Script Assist helps you avoid the syntax and logic errors a novice user might make. However, to use Script Assist you must become familiar with ActionScript, and know what methods, functions, and variables to use when creating your scripts. To learn about ActionScript, see *Learning ActionScript 2.0 in Adobe Flash* or *Programming ActionScript 3.0*.

For a video tutorial about Script Assist mode, see www.adobe.com/go/vid0131.

For a text tutorial about Script Assist mode, see Use Script Assist Mode on the Flash Tutorials page at www.adobe.com/go/learn_fl_tutorials.

Use Script Assist to write ActionScript

To add an ActionScript 3.0 action to a Flash document, you must attach it to a frame. To add an ActionScript 2.0 (or earlier) action to a Flash document, attach it to a button or movie clip, or to a frame in the timeline.

For a video tutorial about Script Assist mode, see www.adobe.com/go/vid0131.

For a text tutorial about Script Assist mode, see Use Script Assist Mode on the Flash Tutorials page at www.adobe.com/go/learn_fl_tutorials.

Start Script Assist mode

1 Select Window > Actions.

2 In the Actions panel, click Script Assist .

In Script Assist mode, the Actions panel changes in the following ways:

- Add (+) functions differently in Script Assist mode. When you select an item from the Actions toolbox or the Add menu , the item is added after the currently selected text block.
- Delete (-) lets you remove the current selection in the Script pane.
- The up and down arrows let you move the current selection in the Script pane upward or downward within the code.
- The Check Syntax , Auto Format , Show Code Hint , and Debug Options buttons and menu items normally visible in the Actions panel are disabled, as they do not apply to Script Assist mode.
- The Insert Target button is disabled unless you are typing in a box. Clicking Insert Target places the resulting code in the current box.

Note: *If the Actions panel contains ActionScript code when you click Script Assist, Flash compiles the code. If there are errors in the code, you cannot use Script Assist until you fix the current code selection. The errors are described in detail in the Compiler Errors panel.*

View a description of an action

- Click a category in the Actions toolbox to display the actions in that category, and click an action.
- Select a line of code in the Script pane.

The description appears at the top of the Actions panel.

Add an action to the Script pane

- Click a category in the Actions toolbox to display the actions in that category, and then either double-click an action or drag it to the Script pane.
- Click Add (+) and select an action from the pop-up menu.
- Press Escape and a shortcut key. (To view a list of shortcut keys, select Esc Shortcut Keys in the Actions panel pop-up menu; select this option again to hide the list.)

Delete an action

1 Select a statement in the Script pane.

2 Click Delete (-) or press the Delete key.

Move a statement up or down in the Script pane

1 Select a statement in the Script pane.

2 Click the up or down arrow.

Work with parameters

1 Add an action to, or select a statement in, the Script pane.

Relevant parameter options appear above the Script pane.

2 Enter values in the boxes above the Script pane.

Search for text in a script

- To go to a specific line in a script, choose Go To Line from the Actions panel pop-up menu or press Control+G (Windows) or Command+G (Macintosh); then enter the line number.
- To find text, click Find, select Find from the Actions panel pop-up menu, or press Control+F (Windows) or Command+F (Macintosh).
- To find text again, press F3 or select Find Again from the Actions panel pop-up menu.
- To replace text, click Find, or press Control+H (Windows) or Command+H (Macintosh).

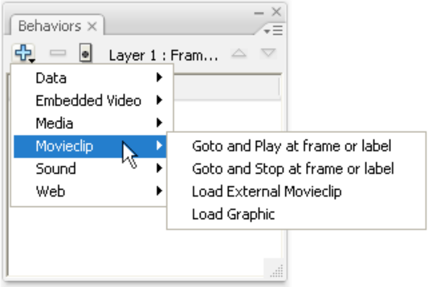
In Script Assist mode, Replace searches and replaces text only in the parameter box of each action, not in the entire script. For example, in Script Assist mode you cannot replace all `gotoAndPlay` actions with `gotoAndStop`.

About behaviors

Behaviors are predefined scripts that you can attach to objects in your FLA file. Behaviors provide functionality such as frame navigation, loading external SWF and JPEG files, controlling the stacking order of movie clips, and movie clip dragging.

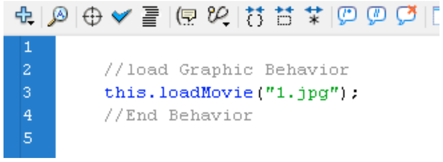
Behaviors provide a convenient way to avoid writing ActionScript and can help you learn how ActionScript works.

Behaviors are available only for ActionScript 2.0 and earlier, and only when you work in the Actions panel, not in an external script file. Typically, you select a triggering object in your document (such as a movie clip or a button), select Add in the Behaviors panel, and then select behavior, as the following example shows:



Selecting a behavior from the Behaviors panel

The behavior is added to the object and is displayed in the Actions panel.



```

    //load Graphic Behavior
    this.loadMovie("1.jpg");
    //End Behavior

```

The ActionScript of a behavior

For a sample about behaviors, see the Flash Samples page at www.adobe.com/go/learn_fl_samples. Download and decompress the Samples zip file and navigate to the Behaviors\BehaviorsScrapbook folder to access the sample.

See also

“Controlling instances with behaviors” on page 224

“Add and configure a behavior” on page 225

“Create custom behaviors” on page 226

Writing and managing scripts

Use code hints

When you work in the Actions panel or Script window, the software can detect what action you are entering and display a *code hint*. There are two types of code hint: a tooltip that contains the complete syntax for that action, and a pop-up menu that lists possible ActionScript elements, such as method or property names (sometimes referred to as a form of *code completion*).

Code hints are enabled by default. By setting preferences, you can disable code hints or determine how quickly they appear. When code hints are disabled in preferences, you can still manually display a code hint for a specific command.

***Note:** If you cannot display code hints for a variable or object you created in ActionScript 2.0, but code hints are enabled in ActionScript preferences, make sure you named your variable or object with the proper suffix or that you used strict typing for your variable or object.*

Specify preference settings for automatic code hints

- In the Actions panel or Script window, select Edit > Preferences (Windows) or Flash > Preferences (Macintosh), click ActionScript in the Category list, and then enable or disable Code Hints.
- Select Preferences from the panel menu ☰ (at the upper right of the Actions panel), and enable or disable Code Hints in the ActionScript preferences.

Specify a delay for code hints

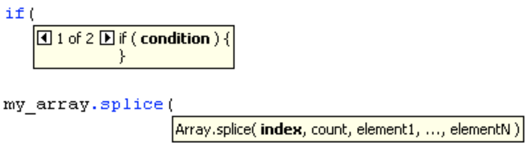
1 In the Actions panel or Script window, select Edit > Preferences (Windows) or Flash > Preferences (Macintosh).

2 Click ActionScript in the Category list.

3 Use the slider to select an amount of delay, in seconds.

Use tooltip-style code hints

1 Display the code hint by typing an opening parenthesis “(“ after an element that requires parentheses (for example, after a method name, a command such as `if` or `do..while`, and so on).



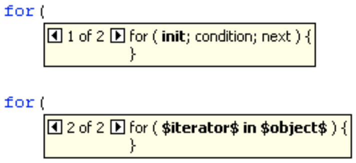
```
if(
   ◀ 1 of 2 ▶ if ( condition ) {
              }

my_array.splice(
                Array.splice( index, count, element1, ..., elementN )
```

Opening parentheses invoke the code hints

2 Enter a value for the parameter.

For multiple parameters, separate the values with commas. For functions or statements, such as the `for` loop, separate the parameters with semicolons.

Overloaded commands (functions or methods that can be invoked with different sets of parameters) such as `gotoAndPlay()` or `for` display an indicator that lets you select the parameter you want to set. To select the parameter, click the small arrows or press Control+Left Arrow and Control+Right Arrow.



```
for (
    1 of 2  for ( init; condition; next ) {
            }

for (
    2 of 2  for ( $iterator$ in $object$ ) {
            }
```

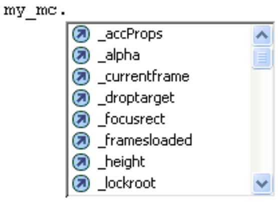
A code hint with multiple sets of parameters

3 To dismiss the code hint, do one of the following:

- Type a closing parenthesis “)”.
- Click outside the statement.
- Press Escape.

Use menu-style code hints:

1 Display the code hint by typing a period after a variable or object name.



```
my_mc.
```

Menu-style code hints

2 To navigate through the code hints, use the Up and Down Arrow keys.

3 To select an item in the menu, press Enter or Tab, or double-click the item.

4 To dismiss the code hint, do one of the following:

- Select one of the menu items.
- Click above or below the menu window.
- Type a closing parenthesis “)” if you’ve already typed an opening parenthesis “(“.
- Press Escape.

Manually display a code hint

1 Click in a code location where code hints can appear, such as in the following locations:

- After the dot (.) following a statement or command, where a property or method must be entered
- Between parentheses [()] in a method name

2 Do one of the following:

- Click Show Code Hint (💬 in the Actions panel or Script window toolbar.

- Press Control+Spacebar (Windows) or Command+Spacebar (Macintosh).
- From the panel menu ▾☰ (at the upper-right corner of the Actions panel), select Show Code Hint.

Reload code hints without restarting the software

❖ From the panel menu ▾☰ (at the upper-right corner of the Actions panel), select Reload Code Hints.

You might need to do this if you customize Script Assist mode by writing custom methods.

Triggering code hints

You can trigger code hints in various ways.

Strict typing for objects

When you use ActionScript 2.0 and use strict typing for a variable that is based on a built-in class (such as Button, Array, and so on), the Script pane displays code hints for the variable. For example, suppose you enter the following two lines of code:

```
var foo:Array = new Array();
foo.
```

As soon as you enter the period (`.`), Flash displays a list of methods and properties available for Array objects in a pop-up menu, because you typed the variable as an array.

Suffixes and code hints

If you use ActionScript 1.0 or you want to display code hints for objects without strictly typing them, add a suffix to the name of each object when you create it. For example, the suffix that triggers code hinting for the Camera class is `_cam`. Suppose you type the following code:

```
var my_array = new Array();
var my_cam = Camera.get();
```

If you then enter `my_cam` followed by a period, code hints for the Camera object appear.

For objects that appear on the Stage, use the suffix in the Instance Name box in the Property inspector. For example, to display code hints for MovieClip objects, use the Property inspector to assign instance names with the `_mc` suffix to all MovieClip objects. Then, whenever you type the instance name followed by a period, code hints appear.

Although suffixes are not required for triggering code hints when you use strict typing for an object, using suffixes consistently helps make your code understandable.

The following table lists the suffixes that trigger code hints in ActionScript 2.0:

Object type	Variable suffix
Array	_array
Button	_btn
Camera	_cam
Color	_color
ContextMenu	_cm
ContextMenuItem	_cmi
Date	_date

Object type	Variable suffix
Error	_err
LoadVars	_lv
LocalConnection	_lc
Microphone	_mic
MovieClip	_mc
MovieClipLoader	_mcl
PrintJob	_pj
NetConnection	_nc
NetStream	_ns
SharedObject	_so
Sound	_sound
String	_str
TextField	_txt
TextFormat	_fmt
Video	_video
XML	_xml
XMLNode	_xmlnode
XMLSocket	_xmlsocket

Comments and code hints

You can also use ActionScript comments to specify an object’s class for code hints. In the following example, a comment tells ActionScript that the class of the `theObject` instance is Object, and so on.

```
// Object theObject;// Array theArray;// MovieClip theMC;
```

If you subsequently enter `theMC` followed by a period, code hints that display the list of MovieClip methods and properties appear. If you enter `theArray` followed by a period, code hints that display the list of Array methods and properties appear, and so on.

Instead of this technique, however, Adobe recommends that you use strict data typing or suffixes, because these techniques enable code hints automatically and make your code more understandable.

Format code

Your code can be formatted and indented automatically or manually. If you use dynamic font mapping, you ensure that the correct fonts are used for multilingual text.

Set Auto Format options

1 Do one of the following:

- From the panel menu ☰ (at the upper-right corner of the Actions panel), select Preferences.
- In the Script window, select Edit > Preferences (Windows) or Flash > Preferences (Macintosh).

2 In the Preferences dialog box, select Auto Format.

3 Select any of the Auto Format options.

After you set Auto Format options, your settings are applied automatically to the code you write, but not to existing code; you must apply your settings to existing code manually.

Format code according to Auto Format settings

- Click Auto Format in the Actions panel or Script window toolbar.
- From the panel menu (at the upper-right corner of the Actions panel), select Auto Format.
- Press Control+Shift+F (Windows) or Command+Shift+F (Macintosh).
- In the Script window, select Tools > Auto Format.

Use dynamic font mapping

❖ To turn dynamic font mapping on or off, select or deselect Use Dynamic Font Mapping in the Preferences dialog box.

Dynamic font mapping is turned off by default because it increases performance time when you are scripting. If you are working with multilingual text, turn on dynamic font mapping because it helps to ensure that the correct fonts are used.

Use automatic indentation

❖ To turn automatic indentation on or off, select or deselect Automatic Indentation in the Preferences dialog box.

When automatic indentation is turned on, the text you type after an opening parenthesis “(“ or opening curly brace “{“ is automatically indented according to the Tab Size setting in ActionScript preferences.

In your scripts, you can indent a line by selecting it and pressing Tab. To remove the indent, select the line and press Shift+Tab.

Comment sections of code

Code comments are parts of code that the ActionScript compiler ignores. A comment lines explains what your code is doing, or temporarily deactivates code that you don’t want to delete. Comment a line of code by beginning it with a double slash (//). The compiler ignores all text on that line after the double slash. You can also comment larger blocks of code by placing a slash and asterisk (/*) at the beginning of the block and an asterisk and slash (*/) at the end of the block.

You can type these comment markers manually, or you can use buttons at the top of the Actions panel or Script window to add them.

Comment a line of code

1 Place the insertion point at the beginning of the line or at the character where you want the comment to begin.

2 Click Apply Line Comment at the top of the Actions panel or Script window.

A double slash (//) is placed at the insertion point.

Comment multiple lines of code

1 Select the lines to comment. (The first and last lines of the selection can be partial lines.)

2 Click Apply Block Comment at the top of the Actions panel or Script window.

Block comment characters are placed at the beginning (/*) and end (*/) of the selection.

Remove a comment

1 Place the insertion point in the line that contains the comment, or select the block of code that is commented.

2 Click Remove Comment.

Use syntax coloring

In ActionScript, as in any language, *syntax* is the way elements are put together to create meaning. If you use incorrect ActionScript syntax, your scripts cannot work.

To highlight syntax errors, set a color-code for parts of your scripts. For example, suppose you set the syntax coloring preference to make keywords appear in blue. When you type `var`, the word `var` appears in blue. However, if you mistakenly type `vae`, the word `vae` remains black, which shows that you made a typing error.

Do one of the following:

- Select Edit > Preferences (Windows) or Flash > Preferences (Macintosh), click ActionScript in the Category list, and specify Syntax Coloring settings.
- From the panel menu (at the upper-right corner of the Actions panel), select Preferences and specify Syntax Coloring settings.
- With the insertion point in the Script pane, press Control-U (Windows) or Command-U (Macintosh).

***Note:** When you write scripts in the Actions panel, commands that are not supported by the version of the player you are targeting appear in yellow in the Actions toolbox. For example, if the Flash Player SWF file version is set to Flash 7, ActionScript that only Flash Player 8 supports appears in yellow in the Actions toolbox.*

See also

“Set ActionScript preferences” on page 385

Use line numbers and word wrap

When you edit or modify code, line numbers make code easier to scroll and parse. Word wrap helps you avoid horizontally scrolling long lines of code (especially when you work in the authoring environment, or at low screen resolutions).

Enable or disable line numbers

- From the panel menu (at the upper-right corner of the Actions panel), select Line Numbers.
- In the Script window, select View > Line Numbers.
- Press Control+Shift+L (Windows) or Command+Shift+L (Macintosh).

Highlight a specific line

- From the panel menu (at the upper-right corner of the Actions panel), select Go To Line.
- In the Script window, select Edit > Go To Line.

Enable or disable line word wrap

- From the panel menu (at the upper-right corner of the Actions panel), select Word Wrap.
- In the Script window, select View > Word Wrap.
- Press Control+Shift+W (Windows) or Command+Shift+W (Macintosh).

Collapse sections of code

To make your code more readable and easier to navigate during programming and debugging, collapse sections of code into a single line. By collapsing sections that you don't need to look at, you can focus on the code you are writing or debugging.

Collapse selected code

1 Select the code to collapse.

2 Click Collapse Selection .

Collapse code between braces or parentheses

1 Place the insertion point inside the braces or parentheses.

2 Click Collapse Between Braces .

Expand collapsed code

❖ Click the plus sign (+) that appears to the left of the collapsed code. (To recollapse the code block, click the minus sign (-) that appears.)

Expand all collapsed code in the current script

❖ Click Expand All .

Adding ActionScript with shortcut keys

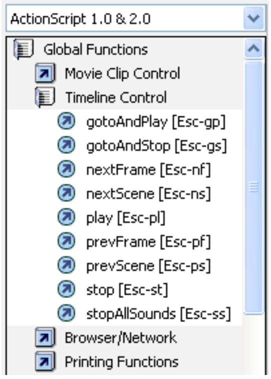
To add elements to a script, use Escape shortcut keys (pressing the Escape key, and then two other keys). For example, if you are working in the Script pane and press Escape+d+o, the following code is placed in your script:

```
do {
} while ();
```

The insertion point is placed after the word `while`, so you can begin typing your condition. Similarly, if you press Escape+c+h, the following code is placed in your script, and the insertion point is placed between the parentheses (), so you can begin typing your condition:

```
catch () {
}
```

To learn which commands have Escape shortcut keys, display them in the Actions toolbox by selecting Escape Shortcut Keys from the Actions Panel menu.



Escape shortcut keys

Display hidden characters

Characters such as spaces, tabs, and line breaks are hidden in ActionScript code. You may need to display these characters; for example, you must find and remove double-byte spaces that are not part of a string value, because they cause compiler errors.

- From the panel menu ▾☰ (at the upper-right corner of the Actions panel), Select Hidden Characters.
- Press Control+Shift+8 (Windows) or Command+Shift+8 (Macintosh).

The following symbols are used to display hidden characters.

Hidden character	**Symbol**
Single-byte space	.
Double-byte space	l
Tab	>>
line break	¶

Find text in a script

The Find tool lets you find and replace text strings in your scripts.

***Note:** To search through text in every script in a Flash document, use the Movie Explorer.*

See also

"Use the Movie Explorer" on page 42

Find text

1 In the Actions panel or Script window, click Find 🔍 or press Control+F (Windows) or Command+F (Macintosh).

2 Enter the search string.

3 Click Find Next.

Find and replace text in a script

1 In the Actions panel or Script window, click Find 🔍 or press Control+F (Windows) or Command+F (Macintosh).

2 Enter the search string.

3 In the Replace box, enter the new string.

4 Click Find Next.

5 To replace the string, click Replace; to replace all occurrences of the string, click Replace All.

Repeat a search in the Actions panel

❖ From the panel menu ▾☰ (at the upper-right corner of the Actions panel), select Find Again.

Repeat a search in the Script window

❖ Select Edit > Find Again.

Check syntax and punctuation

You can do a quick check of your ActionScript code without publishing the FLA file.

When you check syntax, the current script is checked. If the current script calls ActionScript classes, those classes are compiled and their syntax is also checked. Other scripts that might be in the FLA file are not checked.

Check syntax

- In the Actions panel or Script window, click Check Syntax ✓.
- From the panel menu ☰ (at the upper-right corner of the Actions panel), select Check Syntax.
- Click in the Script pane, and then press Control+T (Windows) or Command+T (Macintosh).

Syntax errors are listed in the Compiler Errors panel.

***Note:** In an external ActionScript class file in the Script window, the global classpath affects the syntax check. Even if the global classpath is set correctly, you might generate errors, because the compiler is not aware that this class is being compiled. For more information about ActionScript 3.0 classes, see Classes in Programming ActionScript 3.0. For more information on compiling ActionScript 2.0 classes, see Compiling and exporting classes in Learning ActionScript 2.0 in Adobe Flash.*

Check for punctuation balance

1 Click between braces {}, brackets [], or parentheses () in your script.

2 In Windows, press Control+' (single quote); on the Macintosh, press Command+' (single quote). The text between braces, brackets, or parentheses is highlighted, and you can check that opening punctuation has corresponding closing punctuation.

Encoding for imported and exported scripts

You can set ActionScript preferences to specify the type of encoding to use in imported or exported ActionScript files. UTF-8 Encoding is 8-bit Unicode format, which lets you include text in multiple languages in your file; Default Encoding is the encoding supported by the language your system is currently using, also called the *traditional code page*.

***Important:** When you use a non-English application on an English system, the Test Movie command fails if any part of the SWF file path has characters that cannot be represented by using the Multibyte Character Sets (MBCS) encoding scheme. For example, Japanese paths, which work on a Japanese system, won't work on an English system. Be sure to use English-only path names on English systems. All areas of the application that use the Test Movie player are subject to this limitation.*

Import and export scripts

You can import a script into the Actions panel or Script window. You can also export your scripts from the Actions panel to external ActionScript files. (When you use the Script window, exporting is unnecessary because you can instead save the AS file.)

If text in your scripts doesn't look as expected when you open or import a file, change the import encoding preference.

Import an external AS file

1 In the Script pane, place the insertion point where you want to locate the first line of the external script.

2 Do one of the following:

- In the Actions panel, select Import Script from the panel menu, or press Control+Shift+I (Windows) or Command+Shift+I (Macintosh).
- In the Script window, select File > Import Script or press Control+Shift+I (Windows) or Command+Shift+I (Macintosh).

Export a script from the Actions panel

1 Select the script to export. Then select Export Script from the Actions Panel menu, or press Control+Shift+X (Windows) or Command+Shift+X (Macintosh).

2 Save the ActionScript (AS) file.

Set text encoding options

1 Select Edit > Preferences (Windows) or Flash > Preferences (Macintosh), and select ActionScript in the Category list.

2 Set any of these options:

Open/Import Select UTF-8 Encoding to open or import using Unicode encoding, or select Default Encoding to open or import using the encoding form of the language currently used by your system.

Save/Export Select UTF-8 Encoding to save or export using Unicode encoding, or select Default Encoding to save or export using the encoding form of the language currently used by your system.

Turn the export encoding warning off or on

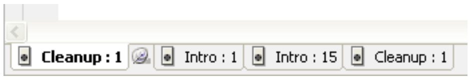
1 Select Edit > Preferences (Windows) or Flash > Preferences (Macintosh), and select Warnings in the Category list.

2 Select or deselect Warn On Encoding Conflicts When Exporting ActionScript Files.

Pin scripts in the Actions panel

If you don't organize the code within your FLA file into one central location, or if you're using behaviors, you can pin individual scripts in the Actions panel to move among them more easily. To *pin* a script means that you can keep the location of the code open in the Actions panel, and easily click between open scripts. This can be especially useful when debugging.

In the following figure, the script associated with the current location on the timeline is on Frame 1 of the layer named Cleanup. (The tab at the far left always follows your location along the timeline.) That script is also pinned (it is shown as the rightmost tab). Two other scripts are pinned: one on Frame 1 and the other on Frame 15 of the layer named Intro. You can move among the pinned scripts by clicking the tabs or by using keyboard shortcuts. Moving among pinned scripts does not change your current position on the timeline.



A pinned script

If the content in the Script pane doesn't change to reflect the location you select on the timeline, the Script pane is probably showing a pinned script. Click the tab at the lower left of the Script pane to show the script associated with your location along the timeline.

See also

"Debugging ActionScript 3.0" on page 410

"Debugging ActionScript 1.0 and 2.0" on page 400

Pin a script

1 Click the Timeline so the script appears in a tab at the lower left of the Script pane in the Actions panel.

2 Do one of the following:

- Click the Pushpin icon to the right of the tab.
- Right-click (Windows) or Control-click (Macintosh) the tab, and select Pin Script.
- From the panel menu ☰ (at the upper-right corner of the Actions panel), select Pin Script.

Unpin a script

- If a pinned script appears in a tab at the lower left of the Script pane in the Actions panel, click the Pushpin icon on the right of the tab.
- Right-click (Windows) or Control-click (Macintosh) a tab, and select Close Script or Close All Scripts.
- From the panel menu ☰ (at the upper-right corner of the Actions panel), select Close Script or Close All Scripts.

Keyboard shortcuts for pinned scripts

When the insertion point is in the Script pane, use the following keyboard shortcuts to work with pinned scripts.

Action	Windows shortcut key	Macintosh shortcut key
Pin script	Control+= (equal sign)	Command+=
Unpin script	Control+- (minus sign)	Command+-
Move focus to tab on the right	Control+Shift+. (period)	Command+Shift+.
Move focus to tab on the left	Control+Shift+, (comma)	Command+Shift+,
Unpin all scripts	Control+Shift+- (minus sign)	Command+Shift+-

Insert target paths

Many script actions are meant to affect movie clips, buttons, and other symbol instances. In your code, you can reference symbol instances on a timeline by inserting a *target path*—the address of the instance you want to target. You can set either an absolute or relative target path. An absolute path contains the entire address of the instance. A relative path contains only the part of the address that is different from the address of the script itself in the FLA file, and will no longer work if the script is moved to another location.

1 In the Actions panel, click an action in your script.

2 Click Target Path ⊕.

3 Enter the path to the target instance, or select the target from the list.

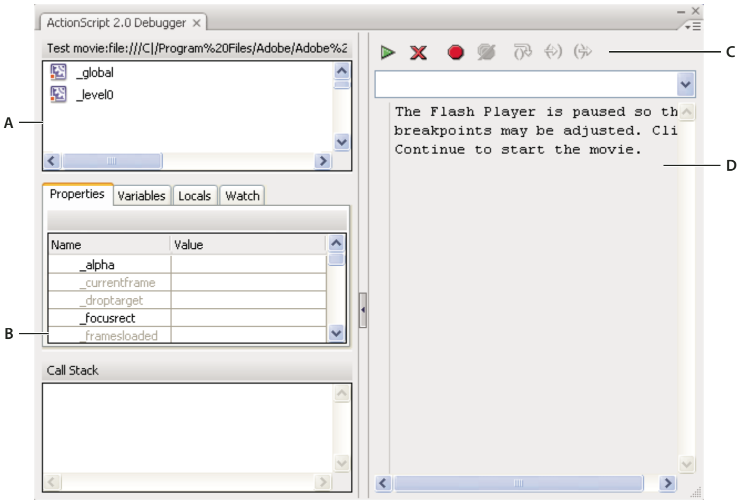
4 Select either the Absolute or Relative path option.

Debugging ActionScript 1.0 and 2.0

Debugging your ActionScript 1.0 and 2.0 scripts

The ActionScript 2.0 debugger helps you find ActionScript 1.0 and 2.0 errors while your SWF file runs in Flash Player. When using Flash to debug ActionScript 1.0 and 2.0, you view your SWF files in the debugger version of Flash Player, which is installed automatically with Flash. To install a stand-alone debugger version of Flash Player, run the installer in the /*Flash installed directory*/Players/Debug/ directory.

The ActionScript 2.0 Debugger shows a hierarchical display list of movie clips currently loaded in Flash Player. Using the ActionScript 2.0 Debugger, you can display and modify variable and property values as the SWF file plays, and you can use breakpoints to stop the SWF file and step through ActionScript code line by line. You can then go back to your scripts and edit them so that they produce the correct results.


ActionScript 2.0 Debugger
Test movie:file:///C|/Program%20Files/Adobe/Adobe%2
_global
_level0
Properties
Variables
Locals
Watch
Call Stack
A
B
C
D

Name	Value
_alpha	
_currentframe	
_droptarget	
_focusrect	
_framesloaded	

```
The Flash Player is paused so th
breakpoints may be adjusted. Cli
Continue to start the movie.
```

***A.** Display list* ***B.** Properties list* ***C.** Tool bar* ***D.** Code view*

You can resize the regions of the Debugger panel. When your pointer changes between each region, drag to resize the Display list, Watch list, and code view. You can also click the vertical bar to expand either side of the Debugger to full size.

After it’s activated, the Debugger status bar displays the URL or local path of the file, tells whether the file is running in the test environment or from a remote location, and shows a live view of the movie clip display list. When movie clips are added to or removed from the file, the display list reflects the changes immediately.

***Note:** The ActionScript 2.0 and 3.0 debuggers have significant differences. For more information about the ActionScript 3.0 debugger, see “Debugging ActionScript 3.0” on page 410.*

The Test Movie command and keyboard controls

When you use the Control > Test Movie command to test SWF files that implement keyboard controls (tabbing, keyboard shortcuts created using `Key.addListener()`, and so on), select Control > Disable Keyboard Shortcuts. Selecting this option prevents the authoring environment from “grabbing” keystrokes, and lets them pass through to the player. For example, in the authoring environment, Control+U opens the Preferences dialog box. If your script assigns Control+U to an action that underlines text onscreen, when you use Test Movie, pressing Control+U opens the Preferences dialog box instead of running the action that underlines text. To let the Control+U command pass through to the player, you must select Control > Disable Keyboard Shortcuts.

***Important:** When you use a non-English application on an English system, the Test Movie command fails if any part of the SWF file path has characters that cannot be represented with the MBCS encoding scheme. For example, Japanese paths on an English system do not work.*

Other debugging tools

Flash also provides the following debugging tools:

- The Compiler Errors panel, which shows errors encountered when Flash compiles your scripts
- The Output panel, which shows runtime error messages, and lists of variables and objects
- The `trace()` statement, which sends programming notes and values of expressions to the Output panel
- The `throw` and `try..catch..finally` statements, which let you test and respond to runtime errors from within your script

Debug a local SWF file

1 Open the FLA document.

2 Select Debug > Debug Movie.

This command exports the SWF file with debugging information (the SWD file). It opens the Debugger and opens the SWF file in the test environment. The SWD file is used to debug ActionScript, and contains information that lets you use breakpoints and step through code.

Debug a remote ActionScript 2.0 SWF file

You can debug a remote SWF file by using the stand-alone, ActiveX, or plug-in version of the Debug Flash Player, which you can find in the *Flash install directory*/Players/Debug/ directory.

To permit remote debugging of the file, enable debugging in the Publish settings. You can also publish your file with a debugging password to ensure that only trusted users can debug it.

As in JavaScript or HTML, users can view client-side variables in ActionScript. To store variables securely, send them to a server-side application instead of storing them in your file. However, as a developer, you may have other trade secrets, such as movie clip structures, that you do not want to reveal. You can use a debugging password to protect your work.

Enable remote debugging of a SWF file and set a debugging password

1 Open the FLA file.

2 Select File > Publish Settings.

3 On the Flash tab of the Publish Settings dialog box, select Permit Debugging.

4 To set a password, enter a password in the Password box.

After you set this password, no one can download information to the Debugger without the password.

5 Close the Publish Settings dialog box, and select one of the following commands:

- Debug > Debug Movie
- File > Export > Export Movie
- File > Publish

Flash creates a debugging file, with the extension .swd, and saves it in the same directory as the SWF file. The SWD file is used to debug ActionScript, and contains information that lets you use breakpoints and step through code.

6 Upload the SWF file and the SWD file to the same directory on your web server, or leave it on the local machine to perform a remote debug session on the localhost.

If the SWD file is not in the same directory as the SWF file, you can still debug remotely; however, the Debugger has no breakpoint information, so you can't step through code.

7 In Flash, select Debug > Begin Remote Debug Session > ActionScript 2.0.

Flash opens the ActionScript 2.0 Debugger panel and waits for a debug Flash Player to connect. You have 2 minutes to start the debug Flash Player. If more than 2 minutes elapse, repeat this step.

8 Open the SWF file in the debug version of the Flash Player plugin, ActiveX control, or stand-alone player. The debug stand-alone player is located in the *Flash install directory*/Players/Debug/ directory.

The debug session begins when the debug player connects to the Flash ActionScript 2.0 Debugger panel.

Activate the Debugger from a remote location

1 Open the Flash authoring application if it is not already open.

2 Select Debug > Begin Remote Debug Session > ActionScript 2.0.

3 In a browser or in the debugger version of the stand-alone player, open the published SWF file from the remote location. Be sure the SWD file is in the same folder as the SWF file.

If the Remote Debug dialog box does not appear, right-click (Windows) or Control-click (Macintosh) in the SWF file to display the context menu, and select Debugger.

4 In the Remote Debug dialog box, select Localhost or Other Machine:

- Select Localhost if the debugger version of Flash Player and the Flash authoring application are on the same computer.
- Select Other Machine if the debugger version of Flash Player and the Flash authoring application are not on the same computer. Enter the IP address of the computer running the Flash authoring application.

5 Enter your debugging password if you set one.

The display list of the SWF file appears in the Debugger. If the SWF file doesn't play, the Debugger might be paused, so click Continue to start it.

Display and modify the values of variables in the Debugger

The Variables tab in the Debugger shows the names and values of any global and timeline variables that are selected in the SWF file's display list. If you change the value of a variable on the Variables tab, the change is reflected in the SWF file while it runs. For example, to test collision detection in a game, you can enter the variable value to position a ball in the correct location next to a wall.

The Locals tab in the Debugger shows the names and values of any local variables that are available in the line of ActionScript where the SWF file is currently stopped, at a breakpoint or anywhere else within a user-defined function.

See also

"List a SWF file's objects and variables" on page 407

Display a variable and its value

1 In the Debugger's display list, select the movie clip containing the variable. (To display global variables, select the `_global` clip in the display list.)

2 Click the Variables tab.

The display list updates automatically as the SWF file plays.

Note: *If a movie clip is removed from the SWF file at a specific frame, that movie clip, along with its variable and variable name, is also removed from the display list in the Debugger. However, if its variable is marked for the Watch list, you can still view it in the Watch tab.*

Modify a variable value

❖ On the Debugger panel Variables tab, double-click the value, and enter a new value.

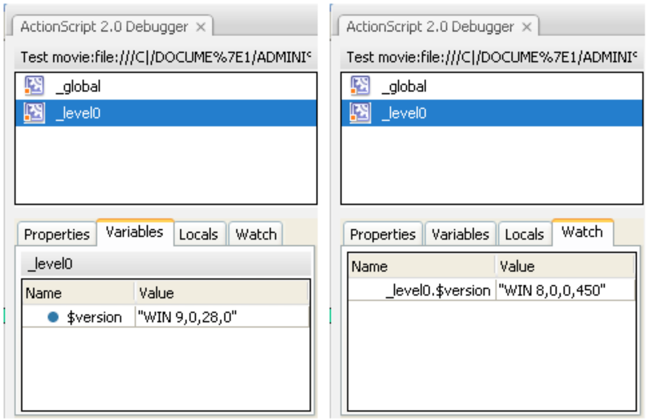
Enter a string (any value surrounded by quotation marks), a number, or a Boolean value (`true` or `false`). You cannot enter an expression (for example, `eval("name:" +i)` or `x + 2`).

Note: *To write the value of an expression to the Output panel in the test environment, use the* `trace()` *statement.*

Use the Watch list

To monitor a set of critical variables, you can mark them for the Watch list. The Watch list shows the absolute path to the variable and the value. You can also enter a new variable value in the Watch list. The Watch list shows only variables that you access by using an absolute target path, such as `_global` or `_root`.

If you add a local variable to the Watch list, its value appears only when Flash Player stops at a line of ActionScript that is within the variable's scope. All other variables appear while the SWF file is playing. If the Debugger can't find the value of the variable, the value is listed as undefined.


ActionScript 2.0 Debugger ×
Test movie:file:///C|/DOCUME%7E1/ADMINI<
_global
_level0
Properties
Variables
Locals
Watch
_level0

Name	Value
● $version	"WIN 9,0,28,0"

ActionScript 2.0 Debugger ×
Test movie:file:///C|/DOCUME%7E1/ADMINI<
_global
_level0
Properties
Variables
Locals
Watch

Name	Value
_level0.$version	"WIN 8,0,0,450"

Variables marked for the Watch list and variables in the Watch list

Add variables to the Watch list

- On the Variables or Locals tab, right-click (Windows) or Control-click (Macintosh) a selected variable; then select Watch from the context menu. A blue dot appears next to the variable.
- On the Watch tab, right-click (Windows) or Control-click (Macintosh) and select Add from the context menu. Double-click in the name column, and enter the target path to the variable name.

Remove variables from the Watch list

❖ On the Watch tab or the Variables tab, right-click (Windows) or Control-click (Macintosh) and select Remove from the context menu.

Display movie clip properties and change editable properties

The Debugger's Properties tab shows all the property values of any movie clip on the Stage. If you change a value, you can see its effect in the SWF file while it runs. (Some movie clip properties are read-only and cannot be changed.)

Display a movie clip's properties in the Debugger

1 Select a movie clip from the display list.

2 Click the Properties tab in the Debugger.

Modify a property value

❖ In the Debugger's Properties tab, double-click the value, and enter a new value.

Enter a string (any value surrounded by quotation marks), a number, or a Boolean value (`true` or `false`). You cannot enter an expression (for example, `x + 50`), or object or array values (for example, `{id: "rogue"}` or `[1, 2, 3]`).

Note: *To write the value of an expression to the Output panel in the test environment, use the* `trace()` *statement.*

Set and remove breakpoints

A breakpoint lets you stop a Flash application at a specific line of ActionScript. You can use breakpoints to test possible trouble spots in your code. For example, if you've written a set of `if..else if` statements and can't determine which one is executing, you can add a breakpoint before the statements and examine them one by one (step through them) in the Debugger.

You can set breakpoints in the Actions panel, Script window, or Debugger. Breakpoints set in the Actions panel are saved with the FLA file. Breakpoints set in the Debugger and Script window are not saved in the FLA file and are valid only for the current debugging session.

Important: *If you set breakpoints in the Actions panel or Script window and click Auto Format, check your breakpoints. If the Auto Format command removed empty lines, your ActionScript might be moved to a different line. It's a good idea to autoformat your scripts before you set breakpoints.*

You can view breakpoints in both the Debugger and the Script window by setting them in either one of those windows. For this to work, the path to the AS file must be the same in both windows.

Do not set breakpoints on comments or empty lines; these breakpoints are ignored.

Set or remove a breakpoint in the Actions panel or Script window

During a debugging session, do one of the following:

- Click in the left margin of the Script pane. A red dot indicates a breakpoint.
- Click Debug Options button.
- Right-click (Windows) or Control-click (Macintosh) to display the context menu, and select Set Breakpoint, Remove Breakpoint, or Remove Breakpoints In This File. (In the Script window, you can also select Remove Breakpoints In All AS Files.)
- Press Control+Shift+B (Windows) or Command+Shift+B (Macintosh).

***Note:** In some previous versions of Flash, clicking in the left margin of the Script pane selected the line of code; now it adds or removes a breakpoint. To select a line of code, use Control-click (Windows) or Command-click (Macintosh).*

Set and remove breakpoints in the Debugger

- Click in the left margin of the code pane. A red dot indicates a breakpoint.
- Click Toggle Breakpoint or Remove All Breakpoints above the code view.
- Right-click (Windows) or Control-click (Macintosh) to display the context menu, and select Set Breakpoint, Remove Breakpoint, or Remove All Breakpoints in the File.
- Press Control+Shift+B (Windows) or Command+Shift+B (Macintosh).

The breakpoints XML file

When you work with breakpoints in the Script window, the AsBreakpoints.xml file lets you store breakpoint information. This file is written to the Local Settings directory, in the following locations:

Windows *Hard Disk*\Documents and Settings*User*\Local Settings\Application Data\Adobe\Flash CS3*language*\Configuration\Debugger\

Macintosh *Macintosh HD*/Users/*User*/Library/Application Support/Adobe Flash CS3/Configuration/Debugger/

Here is an example of an AsBreakpoints.xml file:

```
<?xml version="1.0"?>
<flash_breakpoints version="1.0">
    <file name="c:\tmp\myscript.as">
        <breakpoint line="10"></breakpoint>
        <breakpoint line="8"></breakpoint>
        <breakpoint line="6"></breakpoint>
    </file>
    <file name="c:\tmp\myotherscript.as">
        <breakpoint line="11"></breakpoint>
        <breakpoint line="7"></breakpoint>
        <breakpoint line="4"></breakpoint>
    </file>
</flash_breakpoints>
```

The XML file consists of the following tags:

flash_breakpoints This node has a `version` attribute, which specifies the version of the XML file. Flash 8 is version 1.0.

file A child node of `flash_breakpoints`. This node has a `name` attribute, which specifies the name of the file that contains breakpoints.

breakpoint A child node of `file`. This node has a `line` attribute, which specifies the line number that contains the breakpoint.

The AsBreakpoints.xml file is read when you start Flash, and regenerated when you quit. AsBreakpoints.xml is used to keep track of the breakpoints between development sessions.

Stepping through lines of code

After you set breakpoints in a script and click Continue in the Debugger, you can step through lines of code—that is, control how the Debugger moves through statements and functions.

For example, in the following ActionScript 2.0 code, suppose a breakpoint is set inside a button on the `myFunction()` line:

```
on(press){
    myFunction();
}
```

When you click the button, Flash Player reaches the breakpoint and pauses. You can now bring the Debugger to the first line of `myFunction()` wherever it is defined in the document. You can also continue through or exit out of the function.

As you step through lines of code, the values of variables and properties change in the Variables, Locals, Properties, and Watch tabs. A yellow arrow on the left side of the Debugger's code view indicates the line at which the Debugger stopped. Use the following buttons along the top of the code view:

Step In Advances the Debugger into a function. (If a line does not contain a user-defined function, Step In advances to the next line.)

In the following example, if you place a breakpoint at line 7 and click Step In, the Debugger advances to line 2, and another click of Step In advances you to line 3.

```
1 function myFunction() {
2 x = 0;
3 y = 0;
4 }
5
6 mover = 1;
7 myFunction();
8 mover = 0;
```

***Note:** The numbers in this code snippet denote line numbers. They are not part of the code.*

Step Out Advances the Debugger out of a function. This button works only if you are currently stopped in a user-defined function; it moves the yellow arrow to the line that follows the function call. In the previous example, if you place a breakpoint at line 3 and click Step Out, the Debugger moves to line 8. Clicking Step Out at a line that is not within a user-defined function is the same as clicking Continue. For example, if you stop at line 6 and click Step Out, the player continues to execute the script until it encounters a breakpoint.

Step Over Advances the Debugger over a line of code. This button moves the yellow arrow to the next line in the script. In the previous example, if you are stopped at line 7 and click Step Over, you advance directly to line 8 without stepping through `myFunction()`, although the `myFunction()` code still executes.

Continue Leaves the line at which the player is stopped and continues playing until a breakpoint is reached.

End Debug Session Makes the Debugger inactive but continues to play the SWF file in Flash Player.

Control compiler warnings

You can control the types of compiler warnings that the ActionScript compiler generates in the Compiler Errors panel. When the compiler reports an error, you can double click on the error to navigate to the line of code that caused the error.

1 Select File > Publish Settings.

2 Click the Flash tab.

3 Click the ActionScript Settings button.

4 Select among the Errors options:

- Strict Mode causes warnings to be reported as errors, which means that compilation will not succeed if those errors exist.

- Warnings Mode causes extra warnings to be reported that are useful for discovering incompatibilities when updating ActionScript 2.0 code to ActionScript 3.0.

Output panel overview

When you test a SWF file, the Output panel can show information to help you troubleshoot your SWF file. To show this information, add `trace()` statements to your code or use the List Objects and List Variables commands.

If you use the `trace()` statement in your scripts, you can send specific information to the Output panel as the SWF file runs. This could include notes about the SWF file's status or the value of an expression. For more information, see the `trace()` function in the *ActionScript 2.0 Language Reference*.

Display or hide the Output panel

❖ Select Window > Output or press F2.

Work with the contents of the Output panel

❖ In the Output Panel menu ▾☰, select a command. Here is a partial list:

Copy Copies all the contents of the Output panel to the computer's Clipboard. To copy a selected portion of the output, select the area you want to copy and then select Copy.

Save To File Saves the Output panel contents to a text file.

Filter Level Select None to prevent any information from appearing in the Output panel; select Verbose to send all information to the Output panel.

Navigate to errors in code

When Flash encounters an error in ActionScript code, either during compiling or execution, it reports the error in the Compiler Errors panel. From the Compiler Errors panel, you can navigate to the line of code that caused the error.

❖ Double click the error in the Compiler Errors panel.

List a SWF file's objects and variables

To list a SWF file's objects, use the List Objects command, which is useful for finding the correct target path and instance names. To list a SWF file's variables, use the List Variables command, with which you can find a variable's name and target path.

Selecting the List Objects or List Variables command clears the contents of the Output panel. If you do not want to lose this information, select Save to File from the Output Panel menu before selecting the command.

List a SWF file's objects

In the test environment, the List Objects command shows the level, frame, object type (shape, movie clip, or button), target paths, and instance names of movie clips, buttons, and text fields in a hierarchical list in the Output panel. (It does not show all ActionScript data objects.)

1 If your SWF file is not running in the test environment, select Control > Test Movie.

2 Select Debug > List Objects.

A list of all the objects currently on the Stage appears in the Output panel. The list does not update automatically as the SWF file plays; you must select the List Objects command each time you want to send the information to the Output panel.

List a SWF file's variables in the Output panel

In the test environment, the List Variables command shows a list of all the variables in the SWF file. Global variables (those declared with the `_global` identifier) appear at the top of the List Variables output in a Global Variables section, and each variable has a `_global` prefix.

In addition, the List Variables command shows getter/setter properties—properties that are created with the `Object.addProperty()` method and start `get` or `set` methods. In the Output panel, the value of a getter/setter property is prefixed with `[getter/setter]`. The software determines the value that appears for a getter/setter property by evaluating the `get` function.

The list does not update automatically as the SWF file plays; you must select the List Variables command each time you want to send the information to the Output panel.

1 Create a new FLA document. For example, you might call it **`listvariables.fla`**.

2 Add the following ActionScript 2.0 in the Actions panel:

```
_global.myName = "Buster";
var myCatSays:String = "meow";
var myNum:Number = 313;
var myArray:Array = ["one", "two", "three"];
```

3 Select Control > Test Movie.

4 Select Debug > List Variables in the test environment.

A list of all the variables currently in the SWF file appears in the Output panel. The following example shows the variables that would be output from the code listed above in step 2:

```
Global Variables:
    Variable _global.myName = "Buster"
Level #0:
Variable _level0.$version = "WIN 9,0,29,3"
Variable _level0.myCatSays = "meow"
Variable _level0.myNum = 313
Variable _level0.myArray = [object #1, class 'Array'] [0:"one",
        1:"two",
        2:"three"
    ]
```

Debugging text field objects

To obtain debugging information about TextField objects, you can use the Debug > List Variables command or the Debug > List Objects command in the test environment. When you use Debug > List Variables, the Output panel uses the following conventions to show TextField properties:

- No more than four properties appear on a line.
- A property with a string value appears on a separate line.
- Color properties appear as hexadecimal numbers (0x00FF00).

- The properties appear in the following order: `variable`, `text`, `htmlText`, `html`, `textWidth`, `textHeight`, `maxChars`, `borderColor`, `backgroundColor`, `textColor`, `border`, `background`, `wordWrap`, `password`, `multiline`, `selectable`, `scroll`, `hscroll`, `maxscroll`, `maxhscroll`, `bottomScroll`, `type`, `embedFonts`, `restrict`, `length`, `tabIndex`, `autoSize`.

The Debug > List Objects command lists TextField objects. If an instance name is specified for a text field, the Output panel shows the full target path including the instance name in the following form:

```
Target = "target path"
```

See also

“Control compiler warnings” on page 406

Use the trace statement

A `trace()` statement sends specific information to the Output panel. For example, while testing a SWF file, you can send results to the Output panel when a button is pressed or a frame plays. The `trace()` statement is similar to the JavaScript `alert` statement.

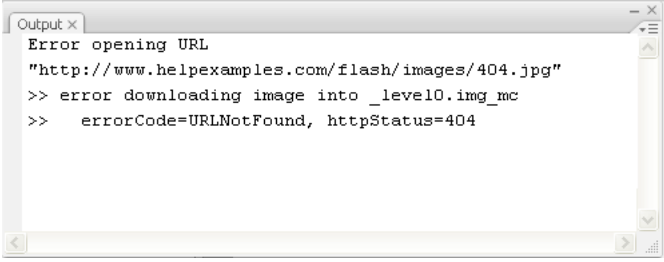
When you use the `trace()` statement in a script, you can use expressions as parameters. The value of an expression appears in the Output panel in the test environment.

1 Select a frame in the Timeline and add a `trace()` statement. For example, you might select Frame 1 and add the following ActionScript 2.0 code:

```
    this.createEmptyMovieClip("img_mc", 10);
    var mclListener:Object = new Object();
    mclListener.onLoadInit = function(target_mc:MovieClip) {
        trace(target_mc+" loaded in "+getTimer()+" ms");
    };
    mclListener.onLoadError = function(target_mc:MovieClip, errorCode:String, 
httpStatus:Number) {
        trace(">> error downloading image into "+target_mc);
        trace(">>\t errorCode="+errorCode+", httpStatus="+httpStatus);
    };
    var img_mcl:MovieClipLoader = new MovieClipLoader();
    img_mcl.addListener(mclListener);
    img_mcl.loadClip("http://www.helpexamples.com/flash/images/404.jpg", img_mc);
```

2 Select Control > Test Movie to test the SWF file.

The Output panel displays the results of the `trace()` statement. For example, the panel might display the following message:



```
Error opening URL
"http://www.helpexamples.com/flash/images/404.jpg"
>> error downloading image into _level0.img_mc
>>   errorCode=URLNotFound, httpStatus=404
```

Output panel

Debugging ActionScript 3.0

About the ActionScript 3.0 debugger

Flash includes a separate debugger for ActionScript 3.0 that operates somewhat differently from the ActionScript 2.0 debugger. The ActionScript 3.0 debugger only works with ActionScript 3.0 FLA and AS files. FLA files must have publish settings set to Flash Player 9. When you initiate an ActionScript 3.0 debugging session, Flash launches the stand-alone debug version of Flash Player to play the SWF file. The debug Flash player plays the SWF in a separate window from the Flash authoring application window.

The ActionScript 3.0 debugger converts the Flash workspace to a debug workspace that displays panels that are used for debugging, including the Actions panel and/or Script window, the Debug console, and the Variables panel. The Debug console displays the call stack and contains tools for stepping through scripts. The Variables panel displays the variables in the current scope with their values and allows you to update those values yourself.

Enter debugging mode

The way you begin a debugging session depends on the type of file you are working on. During a debugging session, Flash interrupts the execution of ActionScript when it encounters a breakpoint or a runtime error.

When Flash initiates a debug session, it adds special information to the SWF file that it exports for the session. This information allows the debugger to provide the specific line numbers in the code where errors are encountered.

You can include this special debugging information in all SWF files created from a specific FLA file in the Publish settings. This allows you to debug the SWF file even if you do not explicitly initiate a debug session. This debugging information makes the SWF file slightly larger.

Start debugging from a FLA file

❖ Select Debug > Debug Movie.

Start debugging from an ActionScript 3.0 AS file

1 With the ActionScript file open in the Script window, select the FLA file that the ActionScript file should be compiled with from the Target menu at the top of the Script window. The FLA file must also be open in Flash to appear in this menu.

2 Select Debug > Debug Movie.

Add debugging information to all SWF files created from a FLA file

1 With the FLA file open, select File > Publish Settings.

2 In the Publish Settings dialog box, click the Flash tab.

3 Select Permit Debugging.

Exit debugging mode

❖ Click the End Debug Session button in the Debug Console.

Set and remove breakpoints

Add breakpoints to ActionScript code to interrupt the execution of the code. After execution is interrupted, you can step through and execute the code line by line, view different sections of your ActionScript, view the values of variables and expressions, and edit variable values.

Note: Breakpoints cannot be added to ASC (ActionScript for Communication) or JSFL (Flash JavaScript) files.

Set a breakpoint

❖ In the Actions panel or Script window, click in the left margin next to the line of code where you want the breakpoint to appear.

Remove a breakpoint

❖ In the Actions panel or Script window, click on the breakpoint to remove.

Step through lines of code

After the ActionScript execution is interrupted at a breakpoint or runtime error, you can step through the code line by line, choosing to step into function calls or step over them. You can also choose to continue executing the code without stepping.

Step into code line by line

❖ Click the Step In button in the Debug Console.

Step over a function call

❖ Click the Step Over button in the Debug Console.

Step out of a function call

❖ Click the Step Out button in the Debug Console.

Resume normal code execution

❖ Click the Continue button in the Debug Console.

Display and examine scripts in the call stack

When code execution stops in the debugger, you can view the call stack in the Debug Console and display the scripts containing the functions in the call stack. The call stack shows the current list of nested function calls that are waiting to complete execution.

You can view the individual scripts that contain each function.

❖ In the Debug Console panel, double click the name of the script in the call stack.

Display and modify variable values

View and edit the values of variables and properties in the Variables panel.

View a variable value

1 In the Variables panel, select the types of variables to display from the Panel menu.

- Show Constants displays the values constants (variables having a fixed value).
- Show Statics displays variables that belong to the class, rather than to instances of the class.
- Show Inaccessible Member Variables displays variables that are not accessible to other classes or namespaces. This includes variables that are protected, private or internal to the namespace.

- Show Additional Hexadecimal Display adds hexadecimal values wherever decimal values are displayed. This is mainly useful for color values. Hexadecimal values are not displayed for decimal values from 0 through 9.
- Show Qualified Names displays variables types with both the package name and the class name.

2 Expand the tree view of the object structure of the FLA until you see the variable to view.

Edit the value of a variable

1 In the Variables panel, double click on the value of the variable.

2 Enter the new value for the variable and press Enter. The new value is used during subsequent code execution.

Control compiler warnings

Control the types of compiler warnings that the ActionScript compiler generates in the Compiler Errors panel. When the compiler reports an error, double click the error to navigate to the line of code that caused the error.

1 Select File > Publish Settings.

2 Click Flash.

3 Click the ActionScript Settings button.

4 Select among the Errors options:

- Strict Mode reports warnings as errors, which means that compilation will not succeed if those errors exist.
- Warnings Mode reports extra warnings that are useful for discovering incompatibilities when updating ActionScript 2.0 code to ActionScript 3.0.

Navigate to errors in code

When Flash encounters an error in ActionScript code, either during compiling or execution, it reports the error in the Compiler Errors panel. Navigate to the line of code that caused the error from the Compiler Errors panel.

❖ Double click the error in the Compiler Errors panel.

Debug a remote ActionScript 3.0 SWF file

With ActionScript 3.0, you can debug a remote SWF file by using the stand-alone, ActiveX, or plug-in version of the Debug Flash Player, which you can find in the *Flash install directory*/Players/Debug/ directory. However, in the ActionScript 3.0 Debugger, remote debugging is limited to files located on the same localhost as the Flash authoring application, being played in the stand alone debug player, ActiveX control, or plugin.

To permit remote debugging of the file, enable debugging in the Publish settings. You can also publish your file with a debugging password to ensure that only trusted users can debug it.

As in JavaScript or HTML, users can view client-side variables in ActionScript. To store variables securely, send them to a server-side application instead of storing them in your file. However, as a developer, you may have other trade secrets, such as movie clip structures, that you do not want to reveal. You can use a debugging password to protect your work.

Enable remote debugging of a SWF file and set a debugging password

In ActionScript 3.0 FLA files, code in frame scripts cannot be debugged. Only code in external AS files can be debugged with the ActionScript 3.0 Debugger.

1 Open the FLA file.

2 Select File > Publish Settings.

3 On the Flash tab of the Publish Settings dialog box, select Permit Debugging.

4 Close the Publish Settings dialog box, and select one of the following commands:

- File > Export > Export Movie
- File > Publish

5 Leave the SWF file on the local machine to perform a remote debug session on the localhost, or upload it to your web server.

The SWF file contains no breakpoint information, so if you upload the file to a remote server you will not be able to step through code. Use the localhost to perform this task.

6 In Flash, select Debug > Begin Remote Debug Session > ActionScript 3.0.

Flash opens the ActionScript 3.0 Debugger and waits for a debug Flash Player to connect. You have 2 minutes to start the debug Flash Player. If more than 2 minutes elapse, repeat this step.

7 Open the SWF file in the debug version of the Flash Player plugin, ActiveX control, or stand-alone player. The debug stand-alone player is located in the *Flash install directory*/Players/Debug/ directory. Do not connect to a file on another machine, as debugger will not be able to receive any breakpoint information.

The debug session begins when the debug player connects to the Flash ActionScript 3.0 Debugger panel.

Activate the Debugger from a remote location

1 Open the Flash authoring application if it is not already open.

2 Select Debug > Begin Remote Debug Session > ActionScript 3.0.

3 In a browser or in the debugger version of the stand-alone player, open the published SWF file from the remote location. Be sure the SWD file is in the same folder as the SWF file.

If the Remote Debug dialog box does not appear, right-click (Windows) or Control-click (Macintosh) in the SWF file to display the context menu, and select Debugger.

4 In the Remote Debug dialog box, select Localhost and select the file to open.

The display list of the SWF file appears in the Debugger. If the SWF file doesn’t play, the Debugger might be paused, so click Continue to start it.

ActionScript publish settings

Modify ActionScript publish settings

When you create a new FLA document, Flash asks you which version of ActionScript you want to use. You can change this setting if you decide later to write your scripts in a different version of ActionScript.

***Note:** The ActionScript 2.0 compiler compiles all ActionScript 1.0 code, except for the slash (/) syntax used to indicate movie clip paths (for example,* `parentClip/testMC:varName= "hello world"`*). To avoid this problem, either rewrite your code using dot (.) notation, or select the ActionScript 1.0 compiler.*

1 Select File > Publish Settings and then select the Flash tab.

2 Select the ActionScript version from the pop-up menu.

Modify the classpath

The classpath tells the ActionScript compiler where to look for external ActionScript files your FLA file references.

When you use ActionScript 2.0, you can set a document-level classpath. This is useful when you create your own classes and you want to override the global ActionScript classpath that is set in the ActionScript preferences.

When you modify a classpath, you can add absolute directory paths (for example, C:/my_classes) and relative directory paths (for example, ../my_classes or ".").

See also

"Set the classpath for ActionScript 2.0" on page 421

"Set the classpath for ActionScript 3.0" on page 422

Modify the global classpath

1 Select Edit > Preferences (Windows) or Flash > Preferences (Macintosh) to open the Preferences dialog box.

2 Click ActionScript in the Category list, and then click ActionScript 2.0 Settings.

3 Do one of the following:

- To add a directory to the classpath, click Browse To Path, browse to the directory you want to add, and click OK.

Alternatively, click Add New Path (+) to add a new line to the Classpath list. Double-click the new line, type a relative or absolute path, and click OK.

- To edit an existing classpath directory, select the path in the Classpath list, click Browse to Path, browse to the directory you want to add, and click OK.

Alternatively, double-click the path in the Classpath list, type the desired path, and click OK.

- To delete a directory from the classpath, select the path in the Classpath list and click Remove From Path.

***Note:** Do not delete the absolute global classpath. The software uses this classpath to access built-in classes. If you accidentally delete this classpath, reinstate it by adding $(LocalData)/Classes as a new classpath for ActionScript 2.0 or $(AppConfig)/ActionScript 3.0/Classes for ActionScript 3.0.*

Modify the document-level classpath

Changing the classpath in the publish settings applies only to the current FLA file.

1 Select File > Publish Settings.

2 In the Publish Settings dialog box, click the Flash tab.

3 Click Settings next to the ActionScript Version pop-up menu.

4 Do one of the following:

- To add a directory to the classpath, click Browse To Path, browse to the directory you want to add, and click OK.

Alternatively, click Add New Path (+) to add a new line to the Classpath list. Double-click the new line, type a relative or absolute path, and click OK.

- To edit an existing classpath directory, select the path in the Classpath list, click Browse To Path, browse to the directory you want to add, and click OK.

Alternatively, double-click the path in the Classpath list, type the desired path, and click OK.

- To delete a directory from the classpath, select the path in the Classpath list, and click Remove From Path.

Declare a document class

When you use ActionScript 3.0, a SWF file may have a top-level class associated with it. This class is called the document class. When the SWF is loaded by Flash Player, an instance of this class is created to be the SWF file's top-level object. This object of a SWF file can be an instance of any custom class you choose.

For example, a SWF file that implements a calendar component can associate its top level with a Calendar class, with methods and properties appropriate to a calendar component. When the SWF is loaded, Flash Player creates it as an instance of this Calendar class.

1 Deselect all objects on the Stage and in the Timeline by clicking a blank area of the Stage. This displays the Document properties in the Property inspector.

2 Enter the path and filename of the ActionScript file for the class in the Document Class text box in the Property inspector.

Note: *You can also enter the Document Class information in the Publish Settings dialog box.*

Customizing context menus in Flash documents

You can customize the standard context menu and the text-editing context menu that appears with SWF files in Flash Player 7 and later.

- The standard context menu appears when a user right-clicks (Windows) or Control-clicks (Macintosh) on a SWF file in Flash Player, in any area except an editable text field. You can add custom items to the menu, and hide any built-in items in the menu except Settings and Debugger.
- The editing context menu appears when a user right-clicks (Windows) or Control-clicks (Macintosh) in an editable text field in a SWF file in Flash Player. You can add custom items to this menu. You cannot hide any built-in items.

Note: *Flash Player also displays an error context menu when a user right-clicks (Windows) or Control-clicks (Macintosh) in Flash Player and no SWF file is loaded. You cannot customize this menu.*

Customize context menus in Flash Player 7 by using the ContextMenu and ContextMenuItem objects in ActionScript 2.0. For more information on using these objects, see `ContextMenu` in the *ActionScript 2.0 Language Reference*.

Remember the following conditions when creating custom context menu items for Flash Player:

- Custom items are added to a context menu in the order in which they are created. You cannot modify this order after the items are created.
- You can specify the visibility and enabling of custom items.
- Custom context menu items are automatically encoded using Unicode UTF-8 text encoding.

Class files and configuration files

When you install Flash, several ActionScript-related configuration folders and files are placed on your system. If you modify these files to configure the authoring environment, back up the original files.

ActionScript classes folder Contains all of the built-in ActionScript 2.0 classes (AS files). Typical paths to this folder are as follows:

- Windows: Hard Disk\Documents and Settings*user*\Local Settings\Application Data\Adobe\Flash CS3*language*\Configuration\Classes

- Macintosh: Hard Disk/Users/*user*/Library/Application Support/Adobe/Flash CS3/*language*/Configuration/Classes

The Classes folder is organized into classes for Flash Player 7 (FP7), classes for Flash Player 8 (FP8), and the mx package, which is used in both players and in ASO files. No separate folder is needed for Flash Player 9. For more information on the organization of this directory, see the Read Me file in the Classes folder.

Include class folder Contains all of the global ActionScript include files. Locations are as follows:

- Windows: Hard Disk\Documents and Settings*user*\Local Settings\Application Data\Adobe\Flash CS3*language*\Configuration\Include
- Macintosh: Hard Disk/Users/*user*/Library/Application Support/Adobe/Flash CS3/*language*/Configuration/Include

ActionsPanel.xml configuration file Includes the configuration file for ActionScript code hinting. Locations are as follows:

- Windows: Hard Disk\Documents and Settings*user*\Local Settings\Application Data\Adobe\Flash CS3*language*\Configuration\ActionsPanel
- Macintosh: Hard Disk/Users/*user*/Library/Application Support/Adobe/Flash CS3/*language*/Configuration/ActionsPanel

AsColorSyntax.xml configuration file The configuration file for ActionScript code color syntax highlighting. Locations are as follows:

- Windows: Hard Disk\Documents and Settings*user*\Local Settings\Application Data\Adobe\Flash CS3*language*\Configuration\ActionsPanel\
- Macintosh: Hard Disk/Users/*user*/Library/Application Support/Adobe/Flash CS3/*language*/Configuration/ActionsPanel

Configuration folders installed with Flash

Flash places several configuration folders on your system when you install the application. The configuration folders organize files associated with the application into appropriate levels of user access. You may want to view the contents of these folders when you are working with ActionScript™ or components. The configuration folders for Flash are as follows:

Application-level configuration folder

Because it is in the application level, nonadministrative users do not have write access to this directory. Typical paths to this folder are as follows:

- In Microsoft Windows XP or Microsoft Windows Vista, browse to *boot drive*\Program Files\Adobe\Adobe Flash CS3*language*\Configuration\.
- On the Macintosh, browse to *Macintosh HD*/Applications/Adobe Flash CS3/Configuration/.

First Run folder

This sibling to the application-level configuration folder facilitates sharing configuration files among users of the same computer. Folders and files in the First Run folder are automatically copied to the user-level configuration folder. Any new files placed in the First Run folder are copied to the user-level configuration folder when you start the application.

Typical paths to the First Run folder are as follows:

- In Windows XP or Vista, browse to *boot drive*\Program Files\Adobe\Adobe Flash CS3*language*\First Run\.
- On the Macintosh, browse to *Macintosh HD*/Applications/Adobe Flash CS3/First Run/.

User-level configuration folder

Found in the user profile area, this folder is always writable by the current user. Typical paths to this folder are as follows:

- In Windows XP or Vista, browse to *boot drive*\Documents and Settings*username*\Local Settings\Application Data\Adobe\Flash CS3*language*\Configuration.
- On the Macintosh, browse to *Macintosh HD*/Users/*username*/Library/Application Support/Adobe/Flash CS3/*language*/Configuration/.

***Note:** Flash for Windows uses the Local Settings folder for the user-level configuration files. This use differs from Flash MX and the rest of the Studio MX family (including Dreamweaver MX 2004), which all use the Roaming Profile configuration folder for user-level configuration files. Roaming profiles allow network users to automatically store their configuration settings on a network server, and then move to different workstations on the network and always have an application load the same configuration files. The Local Settings folder differs from the Roaming Profile folder in that when the network is set up, files in Local Settings are not saved to a server to support roaming profiles.*

All-user-level configuration folder

Found in the common user profile area, this folder is part of the standard Windows and Macintosh operating system installations and is shared by all users of a particular computer. The operating system makes available to all users of the computer any files placed in this folder. Typical paths to this folder are as follows:

- In Windows XP or Vista, browse to *boot drive*\Documents and Settings\All Users\Application Data\Adobe\Flash CS3*language*\Configuration\.
- On the Macintosh, browse to *Macintosh HD*/Users/Shared/Application Support/Adobe/Flash CS3/*language*/Configuration/.

Restricted Users configuration folder

For users with restricted privileges on a workstation, typically, in a networked environment, only system administrators have administrative access to workstations. All other users are given restricted access, which usually means that these users can’t write to application-level files (such as the Program Files directory in Windows or the Applications folder in Macintosh OS X).

Chapter 20: Publishing Flash content

When you're ready to deliver Adobe® Flash® CS3 Professional content to an audience, you can publish it for playback. This chapter describes the various publishing options available with which to distribute your Flash content.

Publishing Flash documents

Publishing overview

By default, the Publish command creates a Flash SWF file, an HTML document that inserts your Flash content in a browser window, and a JavaScript file labeled AC_OETags.js that lets your SWF file play automatically in active content-compliant browsers. The Publish command also creates and copies detection files for Macromedia Flash 4 from Adobe and later. If you change publish settings, Flash saves the changes with the document. After you create a publish profile, export it to use in other documents, or for others working on the same project to use.

Flash Player 6 and later supports Unicode text encoding. With Unicode support, users can view multilanguage text, regardless of the language that the operating system running the player uses.

You can publish the FLA file in alternative file formats—GIF, JPEG, PNG, and QuickTime—with the HTML needed to display them in the browser window. Alternative formats allow a browser to show your SWF file animation and interactivity for users who don't have the targeted Flash Player installed. When you publish a Flash document (FLA file) in alternative file formats, the settings for each file format are stored with the FLA file.

You can export the FLA file in several formats, similar to publishing FLA files in alternative file formats, except that the settings for each file format are not stored with the FLA file.

Alternatively, create a custom HTML document with any HTML editor and include the tags required to display a SWF file.

***Note:** If you create a custom HTML document to display your SWF files in, ensure that it allows your content to play as expected in active content-compliant browsers. For more information about active content, see www.adobe.com/go/activecontent.*

To test how the SWF file works before you publish your SWF file, use Test Movie (Control > Test Movie) and Test Scene (Control > Test Scene).

For video tutorials about publishing Flash content, see the following:

- Publishing Flash content: www.adobe.com/go/vid0141
- Optimizing Flash content: www.adobe.com/go/vid0140

See also

“Using publish profiles” on page 434

“Using Flash Player” on page 419

“Configuring a web server for Flash” on page 432

“Creating multilanguage text” on page 275

Edit a SWF file from Dreamweaver in Flash

If you have both Flash and Dreamweaver installed, you can select a SWF file in a Dreamweaver document and use Flash to edit it. Flash does not edit the SWF file directly; it edits the source document (FLA file) and re-exports the SWF file.

1 In Dreamweaver, open the Property inspector (Window > Properties).

2 In the Dreamweaver document, do one of the following:

- Click the SWF file placeholder to select it; then in the Property inspector click Edit.
- Right-click (Windows) or Control-click (Macintosh) the placeholder for the SWF file, and select Edit With Flash from the context menu.

Dreamweaver switches the focus to Flash, and Flash attempts to locate the Flash authoring file (FLA) for the selected SWF file. If Flash cannot locate the Flash authoring file, you are prompted to locate it.

***Note:** If the FLA file or SWF file is locked, check out the file in Dreamweaver.*

3 In Flash, edit the FLA file. The Flash Document window indicates that you are modifying the file from within Dreamweaver.

4 When you finish making edits, click Done.

Flash updates the FLA file, re-exports it as a SWF file, closes, and then returns the focus to the Dreamweaver document.

***Note:** To update the SWF file and keep Flash open, in Flash select File > Update for Dreamweaver.*

5 To view the updated file in the document, click Play in the Dreamweaver Property inspector or press F12 to preview your page in a browser window.

Using Flash Player

The Flash Player application

Flash Player plays Flash content in the same way as it appears in a web browser or an ActiveX host application. Flash player is installed with the Flash application. When you double-click Flash content, the operating system starts Flash Player, which then plays the SWF file. Use the player to make Flash content viewable for users who aren't using a web browser or an ActiveX host application.

To control Flash content in Flash Player, use menu commands and the `fscommand()` function. For more information, see Sending messages to and from Flash Player in *Learning ActionScript 2.0 in Adobe Flash*.

Use the Flash Player context menu to print Flash content frames.

Playing Flash SWF files

The Flash SWF file format is for deploying Flash content.

You can play content in the following ways:

- In Internet browsers that are equipped with Flash Player
- With Flash Xtra in Director® and Authorware® from Adobe®
- With the Flash ActiveX control in Microsoft Office and other ActiveX hosts

- As part of a QuickTime video
- As a stand-alone application called a projector

The Flash SWF file format is an open standard that other applications support. For more information about Flash file formats, see www.adobe.com/go/flashplayer.

Control applications from Flash Player

❖ Do one of the following:

- To open a new or existing file, select File > New, or Open.
- To change your view of the application, select View > Magnification and make a selection.
- To control Flash content playback, select Control > Play, Rewind, or Loop Playback.

Set publish options for the Flash SWF file format

1 Select File > Publish Settings, click Flash, and select a Player version from the Version pop-up menu. Not all Adobe® Flash® CS3 Professional features work in published SWF files that target Flash Player versions earlier than Flash Player 9. To specify Flash Player detection, click HTML and select Flash Player 4 or later.

2 To specify how Flash loads a SWF file's layers to show the first frame of your SWF file, select a load order (Bottom Up or Top Down). This option controls which parts of the SWF file Flash draws first over a slow network or modem connection.

3 Select the ActionScript™ version from the ActionScript version pop-up menu. If you select ActionScript 2.0 or 3.0 and you've created classes, click Settings to set the relative classpath to class files that differ from the path to default directories set in Preferences.

4 To enable debugging of the published Flash SWF file, select any of the following options:

Generate Size Report Generates a report listing the amount of data in the final Flash content by file.

Omit Trace Actions Causes Flash to ignore Trace actions (`trace`) in the current SWF file. When you select this option, information from Trace actions does not appear in the Output panel. For more information, see "Output panel overview" on page 407.

Protect From Import Prevents others from importing a SWF file and converting it back into a FLA document. Lets you use password protection with your Flash SWF file.

Permit Debugging Activates the Debugger and allows remote debugging of a Flash SWF file. Lets you use password protection with your SWF file.

Compress Movie (Default) Compresses the SWF file to reduce file size and download time. Most beneficial when a file is text- or ActionScript-intensive. A compressed file plays only in Flash Player 6 or later.

Optimize For Flash Player 6 r65 If you selected Flash Player 6 in the Version pop-up menu, select this option to target a release of Flash Player 6. The updated version uses ActionScript register allocation to improve performance. Users must have the same release of Flash Player 6 or later.

Export Hidden Layers (Default) Exports all hidden layers in the Flash document. Deselecting Export Hidden Layers prevents all layers (including layers nested inside movie clips) marked as hidden from being exported in the resulting SWF. This lets you easily test different versions of Flash documents by making layers invisible.

Export SWC Exports a .swc file, which is used for distributing components. The .swc file contains a compiled clip, the component's ActionScript class file, and other files that describe the component.

5 If you are using ActionScript 2.0, and selected either Permit Debugging or Protect From Import, enter a password in the Password text field. If you add a password, other users must enter the password before they can debug or import the SWF file. To remove the password, clear the Password text field. For more information on the Debugger, see “Debugging ActionScript 1.0 and 2.0” on page 400. If you are using ActionScript 3.0, see “Debugging ActionScript 3.0” on page 410.

6 To control bitmap compression, adjust the JPEG Quality slider or enter a value. Lower image quality produces smaller files; higher image quality produces larger files. Try different settings to determine the best trade-off between size and quality; 100 provides the highest quality and least compression.

7 To set the sample rate and compression for all streaming sounds or event sounds in the SWF file, click Set Next To Audio Stream or Audio Event and select options as needed.

***Note:** A streaming sound plays as soon as enough data for the first few frames downloads; it is synchronized to the Timeline. An event sound does not play until it downloads completely, and it continues to play until explicitly stopped.*

8 To override settings for individual sounds selected in the Sound section of the Property inspector, select Override Sound Settings. To create a smaller, low-fidelity version of a SWF file, select this option.

***Note:** If the Select Override Sound Settings option is deselected, Flash scans all stream sounds in the document (including sounds in imported video) and publishes all stream sounds at the highest individual setting. This can increase file size, if one or more stream sounds has a high export setting.*

9 To export sounds suitable for devices, including mobile devices, instead of the original library sound, select Export Device Sounds. Click OK.

10 Select the Flash security model to use from the Local Playback Security pop-up menu. Specify whether to grant your published SWF file local or network security access. Local Access Only lets the published SWF file interact with files and resources on the local system, but not on the network. Access Network Only lets the published SWF file interact with files and resources on the network, but not on the local system.

See also

“Using publish profiles” on page 434

“Working with sound” on page 291

“Using sounds in Flash Lite” on page 295

Set the classpath for ActionScript 2.0

To use an ActionScript class that you’ve defined, Flash must locate the external ActionScript 2.0 files that contain the class definition. The list of folders in which Flash searches for class definitions is called the *classpath*. Classpaths exist at the global, application, or document level. For more information about classpaths, see Classes in *Learning ActionScript 2.0 in Adobe Flash*.

1 Select File > Publish Settings, and click Flash.

2 Verify that ActionScript 2.0 is selected in the ActionScript Version pop-up menu, and click Settings.

3 Specify the frame where the class definition should reside in the Export Frame for Classes text field.

4 Do any of the following:

- To add a folder to the classpath, click the Browse to Path button ⊕, browse to the folder to add, and click OK.
- To add a new line to the Classpath list, click the Add New Path ✚ button. Double-click the new line, type a relative or absolute path, and click OK.

- To edit an existing classpath folder, select the path in the Classpath list, click the Browse to Path button, browse to the folder to add, and click OK. Alternatively, double-click the path in the Classpath list, type the desired path, and click OK.
- To delete a folder from the classpath, select the path in the Classpath list and click the Remove Selected Path button ▬.

Set the classpath for ActionScript 3.0

To use an ActionScript class that you've defined, Flash must locate the external ActionScript 3.0 files that contain the class definition. The list of folders in which Flash searches for class definitions is called the *classpath*. Classpaths exist at the global or document level. For more information about classpaths, see "Packages" in *Programming ActionScript 3.0*.

1 Select File > Publish Settings, and click Flash.

2 Verify that ActionScript 3.0 is selected in the ActionScript Version pop-up menu, and click Settings. Your Flash Player version must be set to Flash Player 9 to use ActionScript 3.0.

3 Specify the frame where the class definition should reside in the Export Frame for Classes text field.

4 Specify the Errors settings. You can choose either Warnings Mode or Strict Mode. Strict Mode reports warnings as errors, which means that compilation will not succeed if those errors exist. Warnings Mode reports extra warnings that are useful for discovering incompatibilities when updating ActionScript 2.0 code to ActionScript 3.0.

5 (Optional) Select Stage to automatically declare stage instances.

6 Specify ActionScript 3.0 or ECMAScript as the dialect to use. ActionScript 3.0 is recommended.

7 Do any of the following:

- To add a folder to the classpath, click the Browse To Path button ⌖, browse to the folder to add, and click OK.
- To add a new line to the Classpath list, click the Add New Path ✚ button. Double-click the new line, type a relative or absolute path, and click OK.
- To edit an existing classpath folder, select the path in the Classpath list, click the Browse To Path button, browse to the folder to add, and click OK. Alternatively, double-click the path in the Classpath list, type the desired path, and click OK.
- To delete a folder from the classpath, select the path in the Classpath list and click the Remove From Path button ▬.

Specify settings that create HTML documents with embedded Flash content

Playing Flash content in a web browser requires an HTML document that activates the SWF file and specifies browser settings. The Publish command automatically generates this document, from HTML parameters in a template document.

The template document can be any text file that contains the appropriate template variables—including a plain HTML file, a file that includes code for special interpreters such as ColdFusion or Active Server Pages (ASP), or a template included with Flash.

To manually enter HTML parameters for Flash or customize a built-in template, use an HTML editor.

HTML parameters determine where the content appears in the window, the background color, the size of the SWF file, and so on, and set attributes for the `object` and `embed` tags. Change these and other settings in the HTML panel of the Publish Settings dialog box. Changing these settings overrides options you've set in the SWF file.

See also

“Configuring a web server for Flash” on page 432

“HTML publishing templates” on page 435

“Editing Flash HTML settings” on page 439

“Using publish profiles” on page 434

“Use device fonts” on page 274

Specify the settings

1 Select File > Publish Settings and click Formats. The HTML file type is selected by default.

2 Use the default filename, which matches the name of your document, or enter a unique name, including the .html extension.

3 To show HTML settings and select an installed template to use from the Template pop-up menu, click HTML. To show a description of the selected template, click Info. The default selection is Flash Only.

4 If you selected an HTML template other than Image Map or QuickTime, and on the Flash tab you set the Version to Flash Player 4 or later, select Flash Version Detection.

Note: *Flash Version Detection configures your document to detect the version of Flash Player that the user has and sends the user to an alternative HTML page if the user does not have the targeted player.*

5 Select a Dimensions option to set the values of the `width` and `height` attributes in the `object` and `embed` tags:

Match Movie (Default) Uses the size of the SWF file.

Pixels Enters the number of pixels for the width and height.

Percent Specifies the percentage of the browser window that the SWF file occupies.

6 To control the SWF file’s playback and features, select Playback options:

Paused At Start Pauses the SWF file until a user clicks a button or selects Play from the shortcut menu. (Default) The option is deselected and the content begins to play as soon as it is loaded (the `PLAY` parameter is set to `true)`.

Loop Repeats the content when it reaches the last frame. Deselect this option to stop the content when it reaches the last frame. (Default) The `LOOP` parameter is on.

Display Menu Shows a shortcut menu when users Right-click (Windows) or Control-click (Macintosh) the SWF file. To show only About Flash in the shortcut menu, deselect this option. By default, this option is selected (the `MENU` parameter is set to `true)` .

Device Font (Windows only) Substitutes anti-aliased (smooth-edged) system fonts for fonts not installed on the user’s system. Using device fonts increases the legibility of type at small sizes and can decrease the SWF file’s size. This option affects only SWF files that contain static text (text that you create when authoring a SWF file and that does not change when the content appears) set to display with device fonts.

7 To determine the trade-off between processing time and appearance, as described in the following list, select Quality options. These options set the `QUALITY` parameter’s value in the `object` and `embed` tags.

Low Favors playback speed over appearance and does not use anti-aliasing.

Auto Low Emphasizes speed at first but improves appearance whenever possible. Playback begins with anti-aliasing turned off. If Flash Player detects that the processor can handle it, anti-aliasing is automatically turned on.

Auto High Emphasizes playback speed and appearance equally at first but sacrifices appearance for playback speed if necessary. Playback begins with anti-aliasing turned on. If the actual frame rate drops below the specified frame rate, anti-aliasing is turned off to improve playback speed. To emulate the View > Antialias setting, use this setting.

Medium Applies some anti-aliasing but does not smooth bitmaps. Medium produces a better quality than the Low setting but lower quality than the High setting.

High (Default) Favors appearance over playback speed and always uses anti-aliasing. If the SWF file does not contain animation, bitmaps are smoothed; if the SWF file contains animation, bitmaps are not smoothed.

Best Provides the best display quality and does not consider playback speed. All output is anti-aliased and bitmaps are always smoothed.

8 Select a Window Mode option, which controls the HTML `wmode` attribute in the `object` and `embed` tags. The window mode modifies the relationship of the content bounding box or virtual window with content in the `HTML` page as described in the following list:

Window (Default) Does not embed any window-related attributes in the object and embed tags. The background of the content is opaque and uses the HTML background color. The HTML code cannot render above or below the Flash content.

Opaque Windowless Sets the background of the Flashcontent to opaque, obscuring anything under the content. Lets HTML content appear above or on top of content.

Transparent Windowless Sets the background of the Flash content to transparent, allowing the HTML content to appear above and below the content. For browsers that support windowless modes, see the following table.

***Note:** In some instances, complex rendering in Transparent Windowless mode can result in slower animation when the HTML images are also complex.*

9 To position the SWF file window in the browser window, select one of the following HTML Alignment options:

Default Centers the content in the browser window and crops edges if the browser window is smaller than the application.

Left, Right, Top, Or Bottom Align SWF files along the corresponding edge of the browser window and crop the remaining three sides as needed.

10 To place the content within specified boundaries if you've changed the document's original width and height, select a Scale option. The Scale option sets the `SCALE` parameter in the `object` and `embed` tags.

Default (Show All) Shows the entire document in the specified area without distortion while maintaining the original aspect ratio of the SWF files. Borders can appear on two sides of the application.

No Border Scales the document to fill the specified area and keeps the SWF file's original aspect ratio without distortion, cropping the SWF file if needed.

Exact Fit Shows the entire document in the specified area without preserving the original aspect ratio, which can cause distortion.

No Scale Prevents the document from scaling when the Flash Player window is resized.

11 To set how the content is placed within the application window and how it is cropped, select a Flash Alignment option. This option sets the `SALIGN` parameter of the `object` and `embed` tags.

12 To show error messages if tag settings conflict—for example, if a template has code referring to an alternative image that was not specified—select Show Warning Messages.

13 To save the settings with the current file, click OK.

Browsers that support windowless modes

Operating system	Internet Explorer	Netscape	Other
Macintosh OS X 10.1.5 and 10.2	5.1 and 5.2	7.0 and later	• Opera 6 or later • Mozilla 1.0 or later • AOL/Compuserve
Windows	5.0, 5.5, and 6.0	7.0 and later	• Opera 6 and later • Mozilla 1.0 and later • AOL/Compuserve

Configure publish settings for Flash Player detection

Flash Player detection is available only for publish settings set to Flash Player 4 or later, and for SWF files embedded in the Flash Only or Flash HTTPS templates.

***Note:** Flash Player 5 and later are installed on 98% of Internet-connected computers, making Flash Player detection a reasonable method to ensure that end users have the correct version of Flash installed to view your content.*

To detect the presence of Flash Player before allowing a browser to view SWF files that use the following templates, create a separate HTML page with its own SWF file to detect Flash Player before redirecting browsers to the HTML page that contains the Flash content.

The following HTML templates do not support Flash Player detection because the JavaScript in these templates conflicts with the JavaScript used to detect the Flash Player:

- Flash for PocketPC 2003
- Flash with AICC Tracking
- Flash with FSCommand
- Flash with Named Anchors
- Flash with SCORM Tracking

***Note:** Image Map and Quicktime HTML Templates do not support Player detection because they do not embed the Flash Player.*

1 Select File > Publish Settings, and click HTML.

2 Select either the Flash Only or Flash HTTPS template from the Template pop-up menu. These templates support the single-page HTML detection kit. Either of these templates enables the Detect Flash Version check box and the version number text fields.

3 Select the Detect Flash Version check box. Your SWF file is embedded in a web page that includes Flash Player detection code. If the detection code finds an acceptable version of Flash Player installed on the end user's computer, the SWF file plays as designed.

4 (Optional) To specify precise revisions of Flash Player, use the Major Revision and Minor Revision text fields. For example, specify Flash Player version 7.0.2 if it provides a feature specific to displaying your SWF file.

When you publish your SWF file, Flash creates a single HTML page in which to embed the SWF file and the Flash Player detection code. If an end user does not have the version of Flash you've specified to view the SWF file, an HTML page appears with a link to download the latest version of Flash Player.

Set publish settings for GIF files

Use GIF files to export drawings and simple animations for use in web pages. Standard GIF files are compressed bitmaps.

An animated GIF file (sometimes referred to as a GIF89a) offers a simple way to export short animation sequences. Flash optimizes an animated GIF file, storing only frame-to-frame changes.

Flash exports the first frame in the SWF file as a GIF file, unless you mark a different keyframe for export by entering the **#Static** frame label in the Property inspector. Flash exports all the frames in the current SWF file to an animated GIF file unless you specify a range of frames for export by entering the **#First** and **#Last** frame labels in the appropriate keyframes.

Flash can generate an image map for a GIF file to maintain URL links for buttons in the original document. Use the Property inspector to place the frame label #Map in the keyframe in which to create the image map. If you don't create a frame label, Flash creates an image map using the buttons in the last frame of the SWF file. Create an image map only if the `$IM` template variable is present in the template you select.

1 Select File > Publish Settings, click Formats, and select GIF Image.

2 For the GIF filename, use the default filename or enter a new filename with the .gif extension.

3 Click GIF.

Dimensions Enter values for width and height in pixels for the exported bitmap image, or select Match Movie to make the GIF the same size as the SWF file and maintain the aspect ratio of your original image.

Playback Determines whether Flash creates a still (Static) image or an animated GIF (Animation). If you select Animation, select Loop Continuously or enter the number of repetitions.

4 To specify a range of appearance settings for the exported GIF file, select one of the following options:

Optimize Colors Removes any unused colors from a GIF file's color table. This option reduces the file size without affecting image quality, but slightly increases the memory requirements. This option has no effect on an adaptive palette. (An adaptive palette analyzes the colors in the image and creates a unique color table for the selected GIF file.)

Interlace Incrementally shows the exported GIF file in a browser as it downloads. Lets the user see basic graphic content before the file completely downloads and can download the file faster over a slow network connection. Do not interlace an animated GIF image.

Smooth Applies anti-aliasing to an exported bitmap to produce a higher-quality bitmap image and improve text display quality. However, smoothing might cause a halo of gray pixels to appear around an anti-aliased image placed on a colored background, and it increases the GIF file size. Export an image without smoothing if a halo appears or if you're placing a GIF transparency on a multicolored background.

Dither Solids Applies dithering to solid colors as well as gradients.

Remove Gradients (Default is off) Converts all gradient fills in the SWF file to solid colors using the first color in the gradient. Gradients increase the size of a GIF file and are often poor quality. To prevent unexpected results, select the first color of your gradients carefully if you use this option.

5 To determine the transparency of the application's background and the way alpha settings are converted to GIF, select one of the following Transparent options:

Opaque Makes the background a solid color.

Transparent Makes the background transparent.

Alpha Sets partial transparency. Enter a Threshold value between 0 and 255. A lower value results in greater transparency. A value of 128 corresponds to 50% transparency.

6 To specify how pixels of available colors are combined to simulate colors not available in the current palette, select a Dither option. Dithering can improve color quality, but it increases the file size.

None Turns off dithering and replaces colors not in the basic color table with the solid color from the table that most closely approximates the specified color. Turning dithering off can result in smaller files but unsatisfactory colors.

Ordered Provides good-quality dithering with the smallest increase in file size.

Diffusion Provides the best-quality dithering but increases file size and processing time. Works only with the web 216-color palette selected.

7 To define the image's color palette, select one of the following Palette types:

Web 216 uses the standard 216-color, web-safe palette to create the GIF image, for good image quality and the fastest processing on the server.

Adaptive analyzes the colors in the image and creates a unique color table for the selected GIF file. Best for systems displaying thousands or millions of colors; it creates the most accurate color for the image but increases file size. To reduce the size of a GIF file with an adaptive palette, use the Max Colors option to decrease the number of colors in the palette.

Web Snap Adaptive is the same as the Adaptive palette option except it converts similar colors to the web 216-color palette. The resulting color palette is optimized for the image, but when possible, Flash uses colors from the web 216-color palette. This produces better colors for the image when the web 216-color palette is active on a 256-color system.

Custom specifies a palette that you optimized for the selected image. The custom palette is processed at the same speed as the web 216-color palette. To use this option, know how to create and use custom palettes. To select a custom palette, click the Palette folder icon (the folder icon that appears at the end of the Palette text field), and select a palette file. Flash supports palettes saved in the ACT format, that some graphics applications export.

8 To set the number of colors used in the GIF image, if you selected the Adaptive or Web Snap Adaptive palette, enter a value for Max Colors. A smaller number of colors can produce a smaller file but can degrade the colors in the image.

9 Click OK.

See also

"Create an image map" on page 438

"Using publish profiles" on page 434

"Import and export color palettes" on page 186

Specify publish settings for JPEG files

The JPEG format lets you save an image as a highly compressed, 24-bit bitmap. Generally, GIF format is better for exporting line art, and JPEG format is better for images with continuous tones, such as photographs, gradients, or embedded bitmaps.

Flash exports the first frame in the SWF file as a JPEG, unless you mark a different keyframe for export by entering the `#Static` frame label.

1 Select File > Publish Settings, click Formats, and select JPEG Image.

2 For the JPEG filename, either use the default filename, or enter a new filename with the .jpg extension.

3 Click JPEG.

Dimensions Enter values for width and height in pixels for the exported bitmap image, or select Match Movie to make the JPEG image the same size as the Stage and maintain the aspect ratio of your original image.

Quality Drag the slider or enter a value to control the amount of JPEG file compression. The lower the image quality, the smaller the file size, and the reverse. To determine the best compromise between size and quality, try different settings.

***Note:** To change the object's compression setting, use the Bitmap Properties dialog box to set the bitmap export quality per object. The default compression option in the Bitmap Properties dialog box applies the Publish Settings JPEG Quality option.*

Progressive Show Progressive JPEG images incrementally in a web browser, which makes images appear faster when loading with a slow network connection. Similar to interlacing in GIF and PNG images.

4 Click OK.

See also

"Using publish profiles" on page 434

"Set bitmap properties" on page 155

Specify publish settings for PNG files

PNG is the only cross-platform bitmap format that supports transparency (an alpha channel). It is also the native file format for Adobe® Fireworks®.

Flash exports the first frame in the SWF file as a PNG file, unless you mark a different keyframe for export by entering the `#Static frame` label.

1 Select File > Publish Settings, click Formats, and select PNG Image.

2 For the PNG filename, either use the default filename, or enter a new filename with the .png extension.

3 Click PNG.

Dimensions Enter values for width and height in pixels for the exported bitmap image, or select Match Movie to make the PNG image the same size as the SWF file and maintain the aspect ratio of your original image.

Bit depth Set the number of bits per pixel and colors to use in creating the image. The higher the bit depth, the larger the file.

- 8-bits per channel (bpc) for a 256-color image
- 24-bpc for thousands of colors
- 24-bpc with Alpha for thousands of colors with transparency (32 bpc)

4 To specify appearance settings for the exported PNG, select from the following options:

Optimize Colors Removes any unused colors from a PNG file's color table, reducing the file size by 1000 to 1500 bytes without affecting image quality but increasing the memory requirements slightly. Has no effect on an adaptive palette.

Interlace Incrementally shows the exported PNG in a browser as it downloads. Lets the user see basic graphic content before the file completely downloads and might download the file faster with a slow network connection. Do not interlace an animated PNG file.

Smooth Applies anti-aliasing to an exported bitmap to produce a higher-quality bitmap image and improve text display quality. However, smoothing might cause a halo of gray pixels to appear around an anti-aliased image placed on a colored background, and it increases the PNG file size. Export an image without smoothing if a halo appears or if you're placing a PNG transparency on a multicolored background.

Dither Solids Applies dithering to solid colors and gradients.

Remove Gradients (Default is off) Converts all gradient fills in the application to solid colors using the first color in the gradient. Gradients increase the size of a PNG and are often poor quality. To prevent unexpected results, select the first color of your gradients carefully if you use this option.

5 If you selected 8-bpc for Bit Depth, select a Dither option to specify how pixels of available colors are mixed to simulate colors not available in the current palette. Dithering can improve color quality, but it increases file size. Select from the following options:

None Turns off dithering and replaces colors not in the basic color table with the solid color from the table that most closely approximates the specified color. Turning dithering off can produce smaller files but unsatisfactory colors.

Ordered Provides good-quality dithering with the smallest increase in file size.

Diffusion Provides the best-quality dithering but increases file size and processing time. It also works only with the web 216-color palette selected.

6 Select one of the following Palette Types to define the color palette for the PNG image:

Web 216 Uses the standard 216-color, browser-safe palette to create the PNG image, for good image quality and the fastest processing on the server.

Adaptive Analyzes the colors in the image and creates a unique color table for the selected PNG file. Best for systems showing thousands or millions of colors; it creates the most accurate color for the image but results in a file size larger than a PNG created with the web 216-color palette.

Web Snap Adaptive Is the same as the Adaptive palette option except that it converts colors similar to the web 216-color palette. The resulting color palette is optimized for the image, but when possible, Flash uses colors from the web 216-color palette. This produces better colors for the image when the web 216-color palette is active on a 256-color system. To reduce the size of a PNG file with an adaptive palette, use the Max Colors option to decrease the number of palette colors.

Custom Specifies a palette that you optimized for the selected image. The custom palette is processed at the same speed as the web 216-color palette. To use this option, know how to create and use custom palettes. To select a custom palette, click the Palette folder icon (the folder icon that appears at the end of the Palette text field), and select a palette file. Flash supports palettes saved in the ACT format, that leading graphics applications export.

7 If you selected the Adaptive or Web Snap Adaptive palette, enter a value for Max Colors to set the number of colors used in the PNG image. A smaller number of colors can produce a smaller file but might degrade the colors in the image.

8 To select a line-by-line filtering method to make the PNG file more compressible, and experiment with the different options for a particular image, select one of the following Filter options:

None Turns off filtering.

Sub Transmits the difference between each byte and the value of the corresponding byte of the previous pixel.

Up Transmits the difference between each byte and the value of the corresponding byte of the pixel immediately above.

Average Uses the average of the two neighboring pixels (left and above) to predict the value of a pixel.

Path Computes a simple linear function of the three neighboring pixels (left, above, upper left), and selects the neighboring pixel closest to the computed value as a predictor of the color.

Adaptive Analyzes the colors in the image and creates a unique color table for the selected PNG file. Best for systems showing thousands or millions of colors; it creates the most accurate color for the image but results in a file size larger than a PNG created with the web 216-color palette. Reduce the size of a PNG created with an adaptive palette by decreasing the number of colors in the palette.

9 Click OK.

See also

"Using publish profiles" on page 434

"Import and export color palettes" on page 186

Specify publish settings for QuickTime videos

The QuickTime Publish Settings option creates videos in the same QuickTime format installed on your computer.

The Flash document plays in the QuickTime video exactly as it does in Flash Player, retaining all its interactive features. If the document also contains a QuickTime video, Flash copies it to its own track in the new QuickTime file.

The current version of QuickTime Player (as of this writing) supports Flash Player 5 SWF file playback. For best results, Flash content that you export to the QuickTime format should contain only those features that Flash Player 5 supports.

If you try to export Flash Player 6 or later content to the QuickTime format, an error message appears, indicating that the installed version of QuickTime does not support that version of Flash Player. To resolve this issue, select File > Publish Settings > Flash, and select Flash Player 5 from the Version pop-up menu.

If a newer version of the QuickTime Player becomes available that supports Flash Player 6 and later versions, install the updated QuickTime version and publish your document as QuickTime files that target those versions of Flash Player.

1 Select File > Publish Settings, click Formats, and select QuickTime file.

2 For the QuickTime filename, either use the default filename, or enter a new filename with the .mov extension.

3 Click QuickTime.

Dimensions Enter a width and height in pixels for the exported QuickTime video, or select Match Movie to make the QuickTime video the same size as the Flash SWF file and keep its aspect ratio.

4 To control the transparency (alpha) mode of the Flash track in the QuickTime video without affecting any alpha settings in the Flash application, select one of the following Alpha options:

Alpha Transparent Makes the Flash track SWF file transparent and shows any content in tracks behind the Flash track.

Copy Makes the Flash track opaque and masks all content in tracks behind the Flash track.

Auto Makes the Flash track transparent if it is on top of any other tracks, but opaque if it is the bottom or only track in the SWF file.

5 To control where the Flash track plays in the stacking order of the QuickTime video, select one of the following Layer options:

Top Places the Flash track always on top of other tracks in the QuickTime video.

Bottom Places the Flash track always behind other tracks.

Auto Places the Flash track in front of other tracks if Flash objects are in front of video objects in the Flash application, and behind all other tracks if Flash objects are not in front.

Streaming Sound Exports all the streaming audio in the Flash SWF file to a QuickTime sound track, recompressing the audio using the standard QuickTime audio settings. To change these options, click Audio Settings; for more information, see your QuickTime documentation.

Controller Specifies the type of QuickTime controller used to play the exported video.

6 To control how QuickTime plays a video, select one of the following Playback options:

Looping Repeats the video when it reaches the last frame.

Paused at Start Pauses the video until a user clicks a button in the video or selects Play from the shortcut menu. By default, the option is deselected; that is, the video begins to play as soon as it is loaded.

Play Every Frame Shows every frame of the video without skipping to maintain time and does not play sound.

7 To combine the Flash content and imported video content into a single QuickTime video, select File Flatten (Make Self-Contained). Deselecting this option makes the QuickTime video refer to the imported files externally; the video won't work properly if these files are missing.

8 Click OK.

See also

"Using publish profiles" on page 434

Preview the publishing format and settings

The Publish Preview command exports the file and opens the preview in the default browser. If you preview a QuickTime video, Publish Preview starts the QuickTime video Player. If you preview a projector, Flash starts the projector.

❖ Select File > Publish Preview, and select the file format to preview.

Using the current Publish Settings values, Flash creates a file of the specified type in the same location as the FLA file. This file remains in this location until you overwrite or delete it.

See also

"Publishing overview" on page 418

Developing applications for mobile devices

Publishing Flash Lite documents

Adobe® Flash® Lite® lets Flash users create engaging content for mobile phones using the ActionScript scripting language, drawing tools, and templates. For detailed information on authoring for mobile devices, see *Developing Flash Lite Applications* and the Content Development Kits on the Mobile and Devices Development Center at www.adobe.com/go/devnet_devices

***Note:** Depending on the mobile device for which you are developing, certain restrictions can apply to which ActionScript commands and sound formats are supported. For more details, see Mobile Articles on the Mobile and Devices Development Center.*

Adobe also provides Adobe Device Central, a new way to test content created with Adobe products on emulated mobile devices. When creating a new mobile document of any kind, start the creation process from Device Central. Device Central lets you select a target device from the beginning of the development process, and have a clear idea what a device's limitations are.

For a video tutorial about creating mobile content in Flash, see www.adobe.com/go/vid0206.

Using Adobe Device Central with Flash

Device Central enables Flash users to preview how Flash files will look and function on a variety of mobile devices.

In the past, it was difficult for Adobe Flash Lite developers to test the files they created on mobile devices. Testing content could take a significant amount of time, especially manually exporting and testing on target devices and returning to Flash to make necessary changes. Device Central is the next generation of mobile emulation and includes new features such as profile updates, memory and performance options, and custom device sets.

For more information, see the Adobe Device Central help system.

Configuring a web server for Flash

Adding MIME types

When a web server accesses files, the server must properly identify the files as Flash content to display them. If the MIME type is missing or not properly delivered by the server, the browser can show error messages or a blank window with a puzzle-piece icon.

If your server is not properly configured, you (or your server's administrator) must add the SWF file MIME types to the server's configuration files and associate the following MIME types with the SWF file extensions:

- MIME type application/x-shockwave-flash has the .swf file extension.
- MIME type application/futuresplash has the .spl file extension.

If you are administering a server, consult your server software documentation for instructions on adding or configuring MIME types. If you are not administering a server, contact your Internet service provider, web master, or server administrator to add the MIME type information.

If your site is on a MAC OS server, you must also set the following parameters: Action: Binary; Type: SWFL; and Creator: SWF2.

Configure a server for Flash Player

For users to view your Flash content on the web, the web server must be properly configured to recognize SWF files.

Your server may already be configured properly. To test server configuration, see TechNote 4151 on the Adobe Flash Support Center at www.adobe.com/go/tn_4151.

Configuring a server establishes the appropriate Multipart Internet Mail Extension (MIME) types so that the server can identify files with the .swf extension as Flash files.

A browser that receives the correct MIME type can load the appropriate plug-in, control, or helper application to process and properly display the incoming data. If the MIME type is missing or not properly delivered by the server, the browser might display an error message or a blank window with a puzzle piece icon.

- If your site is established through an Internet service provider (ISP), ask the ISP to add this MIME type to the server: application/x-shockwave-flash with the .swf extension.
- If you are administering your own server, see your web server documentation for instructions on adding or configuring MIME types.
- Corporate and enterprise system administrators can configure Flash to restrict Flash Player access to resources in the local file system. Create a security configuration file that limits Flash Player functionality on the local system.

The security configuration file is a text file placed in the same folder as the Flash Player installer. The Flash Player installer reads the configuration file during installation and follows its security directives. Flash Player uses the System object to expose the configuration file to ActionScript.

With the configuration file, disable Flash Player access to the camera or microphone, limit the amount of local storage Flash Player can use, control the auto-update feature, and block Flash Player from reading anything from the user's local hard disk.

For more information about security, see System in *ActionScript 2.0 Language Reference*.

Flash security features

Publishing secure Flash documents

Flash Player 8 and later contain the following features that help you ensure the security of your Flash documents:

Buffer overrun protection

Enabled automatically, this feature prevents the intentional misuse of external files in a Flash document to overwrite a user's memory or insert destructive code such as a virus. This prevents a document from reading or writing data outside the document's designated memory space on a user's system.

Exact domain matching for sharing data between Flash documents

Flash Player 7 and later enforces a stricter security model than earlier versions. The security model changed in two primary ways between Flash Player 6 and Flash Player 7:

Exact domain matching Flash Player 6 lets SWF files from similar domains (for example, `www.adobe.com` and `store.adobe.com`) communicate freely with each other and with other documents. In Flash Player 7, the domain of the data to be accessed must match the data provider's domain *exactly* for the domains to communicate.

HTTPS/HTTP restriction A SWF file that loads by using nonsecure (non-HTTPS) protocols cannot access content loaded by using a secure (HTTPS) protocol, even when both protocols are in exactly the same domain.

For more information about ensuring that content performs as expected with the new security model, see Understanding security in *Learning ActionScript 2.0 in Adobe Flash*.

Local and network playback security

Flash Player 8 and later include a security model that lets you determine the local and network playback security for SWF files that you publish. By default, SWF files are granted read access to local files and networks. However, a SWF file with local access cannot communicate with the network, and the SWF file cannot send files or information to any networks.

Allow SWF files to access network resources, letting the SWF file send and receive data. If you grant the SWF file access to network resources, local access is disabled, protecting information on the local computer from potentially being uploaded to the network.

To select the local or network playback security model for your published SWF files, use the Publish Settings dialog box.

See also

“Set publish options for the Flash SWF file format” on page 420

Using publish profiles

About publish profiles

Publish profiles let you:

- Save a publish settings configuration, export it and import the publish profile to other documents or for others to use.
- Import publish profiles to use in your document.
- Create profiles to publish in several media formats.
- Create a publish profile for in-house use that differs from the way you’d publish the files for a client.
- Create a standard publish profile for your company to ensure files are published uniformly.

Publish profiles are saved at the document rather than application level.

Create a publish profile

1 In the Publish Settings dialog box, click the Create New Profile button ✚.

2 Name the publish profile, and click OK.

3 Specify the publish settings for your document, and click OK.

See also

“Publishing overview” on page 418

Duplicate, modify, or delete a publish profile

❖ From the Current Profile pop-up menu (File > Publish Settings), select the publish profile to use:

- To create a duplicate profile, click the Duplicate Profile button ⧉. Enter the profile name in the Duplicate Name text field, and click OK.
- To modify a publish profile, specify the new publish settings for your document, and click OK.
- To delete a publish profile, click the Delete Profile button 🗑, and click OK.

See also

“Publishing overview” on page 418

Export a publish profile

1 From the Current Profile pop-up menu (File > Publish Settings), select the publish profile to export.

2 Click the Import/Export Profile button, and select Export. Export a publish profile as an XML file for import into other documents.

3 Either accept the default location in which to save the publish profile or browse to a new location, and click Save.

Import a publish profile

Other users can create and export publish profiles, which you can import and select as a publish settings option.

1 Select File > Publish Settings, click Import/Export Profile, and select Import.

2 Browse to the publish profile XML file, and click Open.

HTML publishing templates

About HTML publishing templates

A Flash HTML template is a file that contains static HTML code and flexible template code consisting of a special type of variables (which differ from ActionScript variables). When you publish a SWF file, Flash replaces these variables with the values you select in the HTML tab of the Publish Settings dialog box and produces an HTML page with your SWF file embedded.

Flash includes templates, suitable for most users’ needs, that eliminate the need to manually create an HTML page that displays the SWF file. For example, the Flash Only template is useful for testing your files in a browser. It places the SWF file on the HTML page so that you can view it through a web browser with the Flash Player installed.

To publish a new HTML page, use the same template and change the settings. You can create custom templates using any HTML editor. Creating a template is the same as creating a standard HTML page, except that you replace specific values pertaining to a SWF file with variables that begin with a dollar sign ($).

Flash HTML templates have the following special characteristics:

- A one-line title that appears on the Template pop-up menu on the HTML tab of the Publish Settings dialog box.
- A longer description that appears when you click Info on the HTML tab of the Publish Settings dialog box.
- Template variables beginning with a dollar sign ($) that specify where parameter values should be substituted when Flash generates the output file.

***Note:** Use a backslash and dollar sign (\ $) combination to use a dollar sign for another purpose in the document.*

- HTML `object` and `embed` tags that follow the tag requirements of Microsoft Internet Explorer and Netscape Communicator or Navigator, respectively. To display a SWF file properly on an HTML page, follow these tag requirements. Internet Explorer uses the object HTML tag to open a SWF file; Netscape uses the embed tag.

See also

“Using object and embed tags” on page 440

Customize HTML publishing templates

Modify HTML template variables to create an image map, a text report, or a URL report, or to insert custom values for some of the most common Flash `object` and `embed` parameters (for Internet Explorer and Netscape Communicator or Navigator, respectively).

Flash templates can include any HTML content for your application or even code for interpreters such as ColdFusion and ASP.

Flash templates use the AC_OETags.js file to load the SWF file into the HTML page. This JavaScript file lets users immediately interact with Flash content when it loads into the web page instead of having to first click on the Flash content to activate it.

1 Using an HTML editor, open the Flash HTML template to change. These templates are in the following locations:

- Windows XP or Vista: *boot drive*:\Documents and Settings*user*\Local Settings\Application Data\Adobe\Flash CS3*language*\Configuration\HTML\ The Application Data folder is usually a hidden folder; you might need to change your Windows Explorer settings to see this folder.
- Mac OS X 10.3 and later: *Macintosh HD*/Applications/Adobe Flash CS3/First Run/HTML.

The *boot drive* is the drive from which Windows 2000 or Windows XP boots (usually C:). The *user* is the name of the person logged in to the Windows 2000 or Windows XP operating system. The *language* is set to an abbreviated language name. For example, in the US, *language* is set to “en” for English.

2 Edit the template.

3 Save the template in the same folder that you retrieved it from.

4 To apply the template settings to your SWF file, select File > Publish Settings, click HTML, and select the template you modified. Flash changes only the template variables in the template selected.

5 Select your remaining publish settings, and click OK.

See also

“Publishing overview” on page 418

Using HTML template variables

The following table lists the template variables that Flash recognizes:

Attribute/parameter	**Template variable**
`Template title`	`$TT`
`Template description start`	`$DS`
`Template description finish`	`$DF`
`Flash (SWF file) title`	`$T1`
`Flash (SWF file) title for` search engine metadata	`$TL`
Description for search engine metadata	`$DC`
Metadata XML string for use with search engines	`$MD`
`Width`	`$WI`

Attribute/parameter	Template variable
Height	$HE
Movie	$MO
HTML alignment	$HA
Looping	$LO
Parameters for object	$PO
Parameters for embed	$PE
Play	$PL
Quality	$QU
Scale	$SC
Salign	$SA
Wmode	$WM
Devicefont	$DE
Bgcolor	$BG
Movie text (area to write movie text)	$MT
Movie URL (location of SWF file URL)	$MU
Image width (unspecified image type)	$IW
Image height (unspecified image type)	$IH
Image filename (unspecified image type)	$IS
Image map name	$IU
Image map tag location	$IM
QuickTime width	$QW
QuickTime height	$QH
QuickTime filename	$QN
GIF width	$GW
GIF height	$GH
GIF filename	$GN
JPEG width	$JW
JPEG height	$JH
JPEG filename	$JN

Attribute/parameter	Template variable
PNG width	$PW
PNG height	$PH
PNG filename	$PN

See also

"Editing Flash HTML settings" on page 439

Create an image map

Flash can generate an image map to show any image and maintain the function of buttons that link to URLs. When an HTML template includes the `$IM` template variable, Flash inserts the image map code. The `$IU` variable identifies the name of the GIF, JPEG, or PNG file.

1 In your document, select the keyframe to use for the image map and label it **`#Map`** in the frame Property inspector (Window > Properties > Properties). Use any keyframe with buttons that have attached ActionScript 1.0 or 2.0 `getURL` actions.

If you don't create a frame label, Flash creates an image map using the buttons in the last frame of the SWF file. This option generates an embedded image map, not an embedded SWF file.

2 To select the frame to show the image map, do one of the following:

- For PNG or GIF files, label the frame to appear as **`#Static`**.
- For JPEG, during the publish operation, place the playhead on the frame to be used for display.

3 In an HTML editor, open the HTML template you'll modify.

4 Save your template.

5 Select File > Publish Settings, click Format, select a format for the image map, and click OK.

For example, insert the following code in a template:

```
$IM
<img src=$IS usemap=$IU width=$IW height=$IH BORDER=0>
```

This might produce the following code in the HTML document that the Publish command creates:

```
<map name="mymovie">
<area coords="130,116,214,182" href="http://www.adobe.com">
</map>
<img src="mymovie.gif" usemap="#mymovie" width=550 height=400 border=0>
```

Creating text and URL reports

The `$MT` template variable causes Flash to insert all the text from the current SWF file as a comment in the HTML code. This is useful for indexing the content of a SWF file and making it visible to search engines.

The `$MU` template variable makes Flash generate a list of the URLs that actions in the current SWF file refer to and insert the list at the current location as a comment. This action lets link verification tools detect and verify the links in the SWF file.

Using shorthand template variables

The `$PO` (for `object` tags) and `$PE` (for `embed` tags) template variables are useful shorthand elements. Each variable causes Flash to insert into a template any nondefault values for some of the most common `object` and `embed` parameters, including `PLAY` (`$PL`), `QUALITY` (`$QU`), `SCALE` (`$SC`), `SALIGN` (`$SA`), `WMODE` (`$WM`), `DEVICEFONT` (`$DE`), and `BGCOLOR` (`$BG`).

Embedding search metadata

The `$TL` (SWF file title) and `$DC` (description metadata) template variables let you include search metadata in the HTML. This ability can make the SWF file more visible to search engines, and provide meaningful search results. Use the `$MD` template variable to include the search metadata as an XML string.

Sample HTML template

The following Default.HTML template file in Flash includes many of the commonly used template variables:

```
$TTFlash Only
$DS
Display Adobe Flash Movie in HTML.
$DF
<!DOCTYPE html PUBLIC "-//W3C//DTD XHTML 1.0 Transitional//EN"
"http://www.w3.org/TR/xhtml1/DTD/xhtml1-transitional.dtd">
<html xmlns="http://www.w3.org/1999/xhtml" xml:lang="en" lang="en">
<head>
$CS
<title>$TI</title>
</head>
<body bgcolor="$BG">
<!--url's used in the movie-->
$MU
<!--text used in the movie-->
$MT
<object classid="clsid:d27cdb6e-ae6d-11cf-96b8-444553540000"
codebase="http://fpdownload.adobe.com/pub/shockwave/cabs/flash/swflash.cab#version=7,0,0,0
" width="$WI" height="$HE" id="$TI" align="$HA">
<param name="allowScriptAccess" value="sameDomain" />
$PO
<embed $PEwidth="$WI" height="$HE" name="$TI" align="$HA" allowScriptAccess="sameDomain"
type="application/x-shockwave-flash" pluginspage="http://www.adobe.com/go/getflashplayer"
/>
</object>
</body>
</html>
```

Editing Flash HTML settings

About the HTML document

You need an HTML document to play a SWF file in a web browser and specify browser settings. To control a SWF file, change or enter HTML parameters in an HTML editor or create custom HTML files.

Flash can create the HTML document automatically when you publish a SWF file.

See also

“Publishing overview” on page 418

“HTML publishing templates” on page 435

Using object and embed tags

To display a SWF file in a web browser, an HTML document must use the `object` and `embed` tags with the proper parameters.

***Note:** You can generate an HTML document using the correct object and embed tags using the Publish settings dialog box, and selecting the HTML option. For more information, see “Specify settings that create HTML documents with embedded Flash content” on page 422.*

For `object`, four settings (`height`, `width`, `classid`, and `codebase`) are attributes that appear in the `object` tag; all others are parameters that appear in separate, named `param` tags, as shown in the following example:

```
<object classid="clsid:d27cdb6e-ae6d-11cf-96b8-444553540000" width="100"
height="100"
codebase="http://fpdownload.adobe.com/pub/shockwave/cabs/flash/swflash.cab#version=9,0,0,0
">
<param name="movie" value="moviename.swf">
<param name="play" value="true">
<param name="loop" value="true">
<param name="quality" value="high">
</object>
```

For the `embed` tag, all settings (such as `height`, `width`, `quality`, and `loop`) are attributes that appear between the angle brackets of the opening `embed` tag, as shown in the following example:

```
<embed src="moviename.swf" width="100" height="100" play="true"
loop="true" quality="high"
pluginspage="http://www.adobe.com/shockwave/download/index.cgi?P1_Prod_Version=ShockwaveFl
ash">
</embed>
```

To use both tags, position the `embed` tag before the closing `object` tag, as shown in the following example:

```
<object classid="clsid:d27cdb6e-ae6d-11cf-96b8-444553540000" width="100"
height="100"
codebase="http://fpdownload.adobe.com/pub/shockwave/cabs/flash/swflash.cab#version=9,0,0,0
">
<param name="movie" value="moviename.swf">
<param name="play" value="true">
<param name="loop" value="true">
<param name="quality" value="high">
<embed src="moviename.swf" width="100" height="100" play="true”
loop="true" quality="high"
pluginspage="http://www.adobe.com/shockwave/download/index.cgi?P1_Prod_Version=ShockwaveFl
ash">
</embed>
</object>
```

***Note:** If you use the `object` and `embed` tags, use identical values for each attribute or parameter to ensure consistent playback across browsers. The `swflash.cab#version=9,0,0,0` parameter is optional; only omit this parameter if you don’t want to check for the version number.*

Parameters and attributes

The following tag attributes and parameters describe the HTML code that the Publish command creates. Refer to this list as you write custom HTML to show Flash content. Unless noted, all items apply to both the `object` and `embed` tags. Optional entries are noted. Internet Explorer recognizes parameters used with the `object` tag; Netscape recognizes the `embed` tag. Attributes are used with both the `object` and `embed` tags. When you customize a template, you can substitute a template variable (identified in the Value section for each parameter in the following list) for the value.

***Note:** The attributes and parameters listed in this section are shown in lowercase to comply with the XHTML standard.*

See also

"HTML publishing templates" on page 435

devicefont attribute/parameter

Value

`true | false`

Template variable: `$DE`

Description

(Optional) Specifies whether static text objects are rendered in device fonts, even if the Device Font option is not selected. This attribute applies when the necessary fonts are available from the operating system.

src attribute

Value

`movieName.swf`

Template variable: `$MO`

Description

Specifies the name of the SWF file to be loaded. Applies to the `embed` tag only.

movie parameter

Value

`movieName``.swf`

Template variable: `$MO`

Description

Specifies the name of the SWF file to be loaded. Applies to the `object` tag only.

classid attribute

Value

`clsid:d27cdb6e-ae6d-11cf-96b8-444553540000`

Description

Identifies the ActiveX control for the browser. The value must be entered exactly as shown. Applies to the `object` tag only.

width attribute

Value

`n` or `n%`

Template variable: `$WI`

Description

Specifies the width of the application either in pixels or as a percentage of the browser window.

height attribute

Value

`n` or `n%`

Template variable: `$HE`

Description

Specifies the height of the application either in pixels or as a percentage of the browser window.

Note: *Because Flash applications are scalable, quality doesn't degrade at different sizes if the aspect ratio is maintained. (For example, the following sizes all have a 4:3 aspect ratio: 640 x 480 pixels, 320 x 240 pixels, and 240 x 180 pixels.)*

codebase attribute

Value

```
http://fpdownload.adobe.com/pub/shockwave/cabs/flash/swflash.cab#version=7,0,0,0
```

Description

Identifies the location of the Flash Player ActiveX control so that the browser can automatically download it if it is not already installed. The value must be entered exactly as shown. Applies to the `object` tag only.

pluginspage attribute

Value

```
http://www.adobe.com/shockwave/download/index.cgi?P1_Prod_Version=ShockwaveFlash
```

Description

Identifies the location of the Flash Player plug-in so that the user can download it if it is not already installed. The value must be entered exactly as shown. Applies to the `embed` tag only.

swliveconnect attribute

Value

`true | false`

Description

(Optional) Specifies whether the browser should start Java when loading Flash Player for the first time. The default value is `false` if this attribute is omitted. If you use JavaScript and Flash on the same page, Java must be running for the `fscommand()` function to work. However, if you use JavaScript only for browser detection or another purpose unrelated to `fscommand()` actions, you can prevent Java from starting by setting `SWLIVECONNECT` to `false`. To force Java to start when you are not using JavaScript, explicitly set the `SWLIVECONNECT` attribute to `true`. Starting Java substantially increases the startup time for a SWF file; set this tag to `true` only when necessary. Applies to the `embed` tag only.

Use the `fscommand()` action to start Java from a stand-alone projector file.

play attribute/parameter

Value

`true | false`

Template variable: `$PL`

Description

(Optional) Specifies whether the application begins playing immediately on loading in the web browser. If your Flash application is interactive, let the user initiate play by clicking a button or performing another task. In this case, set the `play` attribute to `false` to prevent the application from starting automatically. The default value is `true` if this attribute is omitted.

loop attribute/parameter

Value

`true | false`

Template variable: `$LO`

Description

(Optional) Specifies whether the content repeats indefinitely or stops when it reaches the last frame. The default value is `true` if this attribute is omitted.

quality attribute/parameter

Value

`low | medium | high | autolow | autohigh | best`

Template variable: `$QU`

Description

(Optional) Specifies the level of anti-aliasing to be used. Because anti-aliasing requires a faster processor to smooth each frame of the SWF file before it is rendered on the viewer’s screen, select one of the following values based on whether your priority is speed or appearance:

Low Favors playback speed over appearance and never uses anti-aliasing.

Autolow Emphasizes speed at first but improves appearance whenever possible. Playback begins with anti-aliasing turned off. If Flash Player detects that the processor can handle it, anti-aliasing is turned on.

Autohigh Initially emphasizes playback speed and appearance equally, but sacrifices appearance for playback speed if necessary. Playback begins with anti-aliasing turned on. If the frame rate drops below the specified frame rate, anti-aliasing is turned off to improve playback speed. Use this setting to emulate the Antialias command (View > Preview Mode > Antialias).

Medium Applies some anti-aliasing and does not smooth bitmaps. It produces a better quality than the Low setting but a lower quality than the High setting.

High Favors appearance over playback speed and always applies anti-aliasing. If the SWF file does not contain animation, bitmaps are smoothed; if the SWF file has animation, bitmaps are not smoothed.

Best Provides the best display quality and does not consider playback speed. All output is anti-aliased, and all bitmaps are smoothed.

The default value for `quality` is `high` if this attribute is omitted.

bgcolor attribute/parameter

Value

`#RRGGBB` (hexadecimal RGB value)

Template variable: `$BG`

Description

(Optional) Specifies the background color of the application. Use this attribute to override the background color setting that the SWF file specifies. This attribute does not affect the background color of the HTML page.

scale attribute/parameter

Value

`showall | noborder | exactfit`

Template variable: `$SC`

Description

(Optional) Defines how the application is placed in the browser window when `width` and `height` values are percentages.

Showall (Default) Makes the entire content visible in the specified area without distortion while maintaining the original aspect ratio of the application. Borders can appear on two sides of the application.

Noborder Scales the content to fill the specified area, without distortion but possibly with some cropping, while maintaining the original aspect ratio of the application.

Exactfit Makes the entire content visible in the specified area without trying to preserve the original aspect ratio. Distortion can occur.

The default value is `showall` if this attribute is omitted (and `width` and `height` values are percentages).

align attribute

Value

`Default | L | R | T | B`

Template variable: `$HA`

Description

Specifies the `align` value for the `object`, `embed`, and `img` tags and determines how the SWF file is positioned within the browser window.

Default Centers the application in the browser window and crops edges if the browser window is smaller than the application.

L, R, T, and B Align the application along the left, right, top, and bottom edge, respectively, of the browser window and crop the remaining three sides as needed.

salign parameter

Value

```
L | R | T | B | TL | TR | BL | BR
```

Template variable: `$SA`

Description

(Optional) Specifies where a scaled SWF file is positioned in the area that the `width` and `height` settings define.

L, R, T, And B Align the application along the left, right, top or bottom edge, respectively, of the browser window and crop the remaining three sides as needed.

TL And TR Align the application to the top-left and top-right corner, respectively, of the browser window and crop the bottom and remaining right or left side as needed.

BL And BR Align the application to the bottom-left and bottom-right corner, respectively, of the browser window and crop the top and remaining right or left side as needed.

If this attribute is omitted, the content is centered in the browser window.

base attribute

Value

base directory or URL

Description

(Optional) Specifies the base directory or URL used to resolve all relative path statements in the SWF file. This attribute is helpful when you keep SWF files in a different folder from your other files.

menu attribute or parameter

Value

```
true | false
```

Template variable: `$ME`

Description

(Optional) Specifies what type of menu appears when the viewer Right-clicks (Windows) or Command-clicks (Macintosh) the application area in the browser.

true shows the full menu, which gives the user several options to enhance or control playback.

false shows a menu that contains only the About Adobe Flash Player 6 option and the Settings option.

The default value is `true` if this attribute is omitted .

wmode attribute or parameter

Value

`Window | Opaque | Transparent`

Template variable: `$WM`

Description

(Optional) Lets you use the transparent Flash content, absolute positioning, and layering capabilities available in Internet Explorer 4.0. For a list of browsers this attribute/parameter supports, see "Publishing Flash documents" on page 418.

Window Plays the application in its own rectangular window on a web page. Window indicates that the Flash application has no interaction with HTML layers and is always the topmost item.

Opaque Makes the application hide everything behind it on the page.

Transparent Makes the background of the HTML page show through all the transparent portions of the application and can slow animation performance.

Opaque windowless and Transparent windowless Both interact with HTML layers, letting layers above the SWF file block out the application. Transparent allows transparency so that HTML layers below the SWF file might show through if a section of the SWF file has transparency; opaque does not.

The default value is `Window` if this attribute is omitted. Applies to `object` only.

allowscriptaccess attribute or parameter

Value

`always | never | samedomain`

Description

Use `allowscriptaccess` to let your Flash application communicate with the HTML page hosting it. The `fscommand()` and `getURL()` operations can cause JavaScript to use the permissions of the HTML page, which can be different from the permissions of your Flash application. This has important implications for cross-domain security.

always Permits scripting operations at all times.

never Forbids all scripting operations.

samedomain Permits scripting operations only if the Flash application is from the same domain as the HTML page.

The default value that all HTML publish templates use is `samedomain`.

SeamlessTabbing parameter

Value

`true | false`

Description

(Optional) Lets you set the ActiveX control to perform seamless tabbing, so that the user can tab out of a Flash application. This parameter works only in Windows with the Flash Player ActiveX control, version 7 and higher.

true (or omitted) Sets the ActiveX control to perform seamless tabbing: After users tab through the Flash application, the next tab keypress moves the focus out of the Flash application and into the surrounding HTML content or to the browser status bar if nothing can have focus in the HTML following the Flash application.

false Sets the ActiveX control to behave as it did in version 6 and earlier: After users tab through the Flash application, the next tab keypress wraps the focus around to the beginning of the Flash application. In this mode, you cannot use the tab key to advance the focus past the Flash application.

Chapter 21: Exporting from Flash

Flash provides many formats that you can use to export flash content for use in other applications.

About exporting from Flash

Adobe® Flash® CS3 Professional lets you create content that can be edited in other applications and export Flash content directly into a single format.

Exporting Flash content, images, and video

Export Flash content

Export commands do not store export settings separately with each file, as does the Publish command. (To create all the files you need to put Flash content on the web, use the Publish command.)

Export Movie exports a Flash document to a still-image format, creates a numbered image file for every frame in the document, and exports the sound in a document to a WAV file (Windows only).

1 Open the Flash document to export, or select the frame or image to export in the current document.

2 Select File > Export > Export Movie, or File > Export > Export Image.

3 Enter a name for the output file.

4 Select the file format and click Save. If the format you selected requires more information, an Export dialog box appears.

5 Set the export options for the format you selected. See "About export file formats" on page 448.

6 Click OK, and then click Save.

See also

"Publishing Flash documents" on page 418

"Creating multilanguage text" on page 275

Update Flash content for Dreamweaver

To add the content to your page, export SWF files directly to a Dreamweaver site. Dreamweaver generates all the needed HTML code. You can start Flash from Dreamweaver to update the content. In Dreamweaver, you can update the Flash document (FLA file) and reexport the updated content automatically.

For more information on working with Dreamweaver, see *Using Dreamweaver* in Dreamweaver Help.

1 In Dreamweaver, open the HTML page that contains the Flash content.

2 Do one of the following:

- Select the Flash content within the HTML page, and click Edit.

- In Design view, press Control (Windows) or Command (Macintosh), and double-click the Flash content.
- In Design view, right-click (Windows) or Control-click (Macintosh) the Flash content, and select Edit with Flash.
- In Design view, in the Site panel, right-click (Windows) or Control-click (Macintosh) the Flash content, and select Open with Flash.

3 If the FLA file for the exported file does not open, the Open File dialog box appears. Navigate to the FLA file, and click Open.

4 If the user used the Change Link Sitewide feature in Dreamweaver, a warning appears. To apply link changes to the Flash content, click OK. To prevent the warning message from appearing when you update the Flash content, click Don't Warn Me Again.

5 Update the FLA file as needed in Flash.

6 To save the FLA file and reexport it to Dreamweaver, do one of the following:

- To update the file and close Flash, click the Done button above the upper-left corner of the Stage.
- To update the file and keep Flash open, select File > Update for Dreamweaver.

About export file formats

Remember the following:

- If the format you selected requires more information, an Export dialog box appears.
- When you save a Flash image as a bitmap GIF, JPEG, PICT (Macintosh), or BMP (Windows) file, the image loses its vector information and is saved with pixel information only. You can edit images exported as bitmaps in image editors such as Adobe® Photoshop®, but you can no longer edit them in vector-based drawing programs.
- When you export a Flash file in the SWF format, text is encoded as Unicode, providing support for international character sets, including double-byte fonts. Macromedia Flash Player 6 and later versions support Unicode encoding.

Flash content is exported as sequences, and images are exported as individual files. PNG is the only cross-platform bitmap format that supports transparency (as an alpha channel). Some non-bitmap export formats do not support alpha (transparency) effects or mask layers.

The following table lists the formats that you can export Flash content and images to:

File type	Extension	Windows	Macintosh
"Adobe Illustrator Sequence and Illustrator Image" on page 449	.ai	•	•
"Animated GIF, GIF Sequence, and GIF Image" on page 450	.gif	•	•
"Bitmap (BMP) Sequence and Bitmap Image" on page 450	.bmp	•	
"DXF Sequence and AutoCAD DXF Image" on page 450	.dxf	•	•

File type	Extension	Windows	Macintosh
"Enhanced Metafile (EMF) Sequence and Image (Windows)" on page 450	.emf	•	
"Encapsulated PostScript (EPS) 3.0 with Preview" on page 450	.eps	•	•
"Flash document (SWF)" on page 451	.swf	•	•
"Flash video (FLV)" on page 451	.flv	•	•
"JPEG Sequence and JPEG Image" on page 451	.jpg	•	•
"PICT Sequence and PICT Image (Macintosh)" on page 451	.pct		•
"PNG Sequence and PNG Image" on page 452	.png	•	•
"Exporting QuickTime" on page 453	.mov	•	•
"WAV audio (Windows)" on page 454	.wav	•	
"Windows AVI (Windows)" on page 454	.avi	•	
"Windows Metafile Image and Windows Metafile Sequence" on page 454	.wmf	•	

Adobe Illustrator Sequence and Illustrator Image

When you export a Flash image as a vector-graphic file (in Adobe® Illustrator® format), you preserve its vector information. You can edit these files in other vector-based drawing programs, but you can't import these images into most page-layout and word-processing programs.

Flash supports exporting Adobe Illustrator 88, 3, 5, and 6 formats. Flash does not support the Photoshop EPS format or EPS files that the Print command generates. Select the Adobe Illustrator version in the Export Adobe Illustrator dialog box.

Versions of the Adobe Illustrator format before version 5 do not support gradient fills, and only version 6 supports bitmaps.

See also

"Import Adobe Illustrator files" on page 140

Animated GIF, GIF Sequence, and GIF Image

The settings are the same as those on the GIF tab in the Publish Settings dialog box, with the following exceptions:

Resolution Set in dots per inch (dpi). To use the screen resolution, enter a resolution or click Match Screen.

Include Export the minimum image area or the full document size.

Colors Set the number of colors that can be used to create the exported image. The color choices are: black and white, 4, 6, 16, 32, 64, 128, or 256-colors; or Standard Color (the standard 216-color, browser-safe palette).

Animation Available for the Animated GIF export format only. Enter the number of repetitions, where 0 repeats endlessly.

See also

"Configure publish settings for Flash Player detection" on page 425

Bitmap (BMP) Sequence and Bitmap Image

Create bitmap images for use in other applications. The Export Bitmap dialog box has the following options:

Dimensions Sets the size of the exported bitmap image in pixels. The size you specify always has the same aspect ratio as your original image.

Resolution Sets the resolution of the exported bitmap image in dots per inch (dpi) and automatically calculates width and height based on the size of your drawing. To set the resolution to match your monitor, select Match Screen.

Color Depth Specifies the bit depth of the image. Some Windows applications do not support the newer 32-bit per channel (bpc) depth for bitmap images; if you have problems using a 32-bpc format, use the 24-bpc format.

Smooth Applies anti-aliasing to the exported bitmap. Anti-aliasing produces a higher-quality bitmap image, but it can create a halo of gray pixels around an image placed on a colored background. Deselect if a halo appears.

DXF Sequence and AutoCAD DXF Image

Export as AutoCAD DXF 10 files, for additional editing in a DXF-compatible application.

This format has no definable export options.

Enhanced Metafile (EMF) Sequence and Image (Windows)

A graphics format available in Windows XP and Windows Vista that saves both vector and bitmap information. EMF supports the curves used in Flash drawings better than the older Windows Metafile format. However, some applications do not support this graphics format.

This format has no definable export options.

Encapsulated PostScript (EPS) 3.0 with Preview

You can export the current frame as an EPS 3.0 file for placement in another application, such as a page layout application. A PostScript printer can print an EPS file. You can include a bitmap preview with the exported EPS file for applications that can import and print the EPS files (such as Microsoft Word and Adobe® PageMaker®) but that can't display them onscreen.

This format has no definable exporting options.

Flash document (SWF)

To place the Flash content in another application, such as Dreamweaver, export the entire document as a SWF file. The same options are available for exporting a document as for publishing the document.

See also

“Publishing Flash documents” on page 418

Flash video (FLV)

Import or export a static video stream with encoded audio. Use with communications applications, such as video conferencing and files that contain screen-share encoded data exported from Adobe’s Macromedia Flash Media Server.

When you export video clips with streaming audio in FLV format, the Streaming Audio settings compress the audio.

Files in the FLV format are compressed with the Sorensen codec.

See also

“Set publish options for the Flash SWF file format” on page 420

“Comparing the On2 VP6 and Sorenson Spark video codecs” on page 310

Export a Flash video file

1 Select the video clip in the Library panel.

2 Select Library > Properties and click Export.

3 Enter a name for the exported file. Select a location to save it to, click Save, and click OK.

JPEG Sequence and JPEG Image

These options match the JPEG Publish Settings options with one exception: Match Screen matches the exported image to the size of the Flash content as it appears on your screen. Match Movie matches the JPEG image to the Flash content and maintains the aspect ratio of the original image.

See also

“Specify publish settings for JPEG files” on page 427

PICT Sequence and PICT Image (Macintosh)

The standard graphics format on the Macintosh; can contain bitmap or vector information. Use the Export PICT dialog box to set the following options:

Dimensions Sets the size of the exported bitmap image specified in pixels. The size you specify always has the same aspect ratio as your original image.

Resolution Sets the resolution in dpi and automatically calculates width and height based on the size of your drawing. To set the resolution to match your monitor, select Match Screen. Bitmap PICT images usually look best onscreen with 72-dpi resolution.

Include Export the minimum image area or the full document size.

Color Depth Designate whether the PICT file is object based or bitmap. Object-based images generally look better when printed, and scaling doesn't affect their appearance. Bitmap PICT images normally look best onscreen and can be manipulated in applications such as Adobe Photoshop. You can also select a variety of color depths with bitmap PICT files.

Include Postscript Available only for an object-based PICT file to include information that optimizes printing on a PostScript printer. This information makes the file larger and some applications might not recognize it.

Smooth Bitmap Available only for bitmap PICT images. Applies anti-aliasing to smooth jagged edges of a bitmap image.

PNG Sequence and PNG Image

The PNG export settings options are similar to the PNG publish settings options (which you can apply as well), with the following exceptions:

Dimensions Sets the size of the exported bitmap image to the number of pixels you enter in the Width and Height fields.

Resolution Enter a resolution in dpi. To use the screen resolution and maintain the aspect ratio of your original image, select Match Screen.

Colors The same as the Bit Depth option in the PNG Publish Settings tab and sets the number of bits per pixel to use in creating the image. For a 256-color image, select 8-bit; for thousands of colors, select 24-bpc; for thousands of colors with transparency (32 bpc) select 24-bpc with Alpha. The higher the bit depth, the larger the file.

Include Select to export the minimum image area or specify the full document size.

Filter Options match those in the PNG Publish Settings tab.

See also

"Specify publish settings for PNG files" on page 428

About QuickTime

Flash provides two methods of exporting Flash documents as QuickTime:

QuickTime export Exports a QuickTime file that can be distributed as streaming video, on a DVD, or used in a video editing application such as Adobe® Premiere®. QuickTime export is intended for users who want to distribute Flash content, such as animation, in the Quicktime video format.

Be aware that the performance of the computer used to export QuickTime video may affect the quality of the video. If Flash is unable to export each frame, it will drop frames, resulting in poor video quality. If you encounter dropped frames, try using either a faster computer with more memory, or reduce the frames per second of the Flash document.

Publish as QuickTime Creates an application with a Flash track in the same QuickTime format installed on your computer. This lets you combine the interactive features of Flash with the multimedia and video features of QuickTime in a single QuickTime 4 movie, which anyone with QuickTime 4 or later can view.

If you import a video clip (in any format) into a document as an embedded file, you can publish the document as a QuickTime movie. If you import a video clip in QuickTime format into a document as a linked file, you can also publish the document as a QuickTime movie.

You must set your Publish settings to Flash 5 or earlier to publish QuickTime with Flash track. This means you cannot use Flash features implemented after Flash 6.

***Note:** Beginning with QuickTime 7.1.3, the use of Flash tracks were disabled by default. To publish a QuickTime file with a Flash track using QuickTime 7.1.3 or later, select Edit > Preferences > QuickTime Preferences > Advanced Media Types within QuickTime, and enable the use of Flash tracks.*

Exports all layers in the Flash document as a single Flash track, unless the document contains an imported QuickTime movie. The imported QuickTime movie remains in QuickTime format in the exported application.

See also

"Specify publish settings for QuickTime videos" on page 430

Exporting QuickTime

For a video tutorial about exporting QuickTime, see www.adobe.com/go/vid0142.

1 Select File > Export > Export Movie.

2 Specify settings for the QuickTime movie to export. By default, QuickTime export creates a movie file using the same dimensions as the source Flash document, and exports the Flash document in its entirety. The Export QuickTime Video dialog box contains the following options:

Dimensions The width and height in pixels for the frames of the QuickTime movie. You can specify only the width or the height; the other dimension is automatically set to maintain the aspect ratio of your original document. To set both the width and the height independently of each other, deselect Maintain Aspect Ratio.

***Note:** If the dimensions of the video is particularly large (for example, 740 x 480 pixels), you may need to change the frame rate of the Flash movie to avoid dropping frames.*

***Note:** The Dimensions you set in the QuickTime Export Settings dialog is for the width and height of the Flash movie exported as video. The dimensions you set in the QuickTime Settings dialog specifies the size of the exported QuickTime movie. If you do not change the size in the later dialog, it remains as "current" so you do not need to change it.*

Ignore stage color Creates an alpha channel using the stage color. The alpha channel is encoded as a transparent track, letting you overlay the exported QuickTime movie on top of other content to alter the background color or scene.

To create a QuickTime video with an alpha channel, you must select a video Compression Type that supports 32 bit encoding with an alpha channel. Codecs that support this are Animation, PNG, Planar RGB, JPEG 2000, TIFF, or TGA. You must also select Million of Color+ from the Compressor/Depth setting. To set the compression type and color depth, click the Setting button in the Video category of the Movie Settings dialog box.

When last frame is reached Exports the entire Flash document as a movie file.

After time has elapsed The duration of the Flash document to export in hours:minutes:seconds:milliseconds.

QuickTime Settings Opens the advanced QuickTime settings dialog box. The Advanced settings let you specify custom QuickTime settings. In general, use the default QuickTime settings, as they provide optimal playback performance for most applications. To modify the QuickTime settings, see the documentation provided with Apple QuickTime Pro for information on the available video parameters.

3 Click export.

See also

"Specify publish settings for QuickTime videos" on page 430

WAV audio (Windows)

Exports only the sound file of the current document to a single WAV file. You can specify the sound format of the new file.

To determine the sampling frequency, bit rate, and stereo or mono setting of the exported sound, select Sound Format. To exclude events sounds from the exported file, select Ignore Event Sounds.

Windows AVI (Windows)

Exports a document as a Windows video but discards any interactivity. Good for opening a Flash animation in a video-editing application. Because AVI is a bitmap-based format, documents that contain long or high-resolution animations can quickly become very large.

The Export Windows AVI dialog box has the following options:

Dimensions Specifies a width and height, in pixels, for the frames of an AVI movie. Specify only the width or the height; the other dimension is automatically set to maintain the aspect ratio of your original document. To set both the width and the height, deselect Maintain Aspect Ratio.

Video Format Selects a color depth. Some applications do not yet support the Windows 32-bpc image format. If this format presents problems, use the older 24-bpc format.

Compress Video Select standard AVI compression options.

Smooth Applies anti-aliasing to the exported AVI movie. Anti-aliasing produces a higher-quality bitmap image, but it can cause a halo of gray pixels to appear around images when placed over a colored background. Deselect if a halo appears.

Sound Format Set the sample rate and size of the sound track, and whether to export in mono or stereo. The smaller the sample rate and size, the smaller the exported file, with a possible trade-off in sound quality.

See also

"Compressing sounds for export" on page 295

Windows Metafile Image and Windows Metafile Sequence

This standard Windows graphics format, which most Windows applications support, yields good results for importing and exporting files.

This format has no definable export options.

See also

"Enhanced Metafile (EMF) Sequence and Image (Windows)" on page 450

Chapter 22: Printing with Flash

You can add printing functionality to your Flash document that lets users print from Flash Player. You can use the ActionScript 2.0 `PrintJob` class, or you can use the `print()` or `printAsBitmap()` ActionScript functions. Users can also access the Flash Player context menu and select the Print command there.

Additionally, users can print from a browser, rather than from Flash Player, by selecting a command such as File > Print from the browser window.

Note: *For information on printing from SWF files at runtime using ActionScript 3.0, see "Printing" in Programming ActionScript 3.0.*

Printing from the Flash authoring tool

Print from Flash documents

To preview and edit your documents, print frames from Adobe® Flash® CS3 Professional documents, or specify frames to be printable from Flash Player by a viewer.

When printing frames from a Flash document, use the Print dialog box to specify the range of scenes or frames to print and the number of copies. In Windows, the Page Setup dialog box specifies paper size, orientation, and various print options—including margin settings and whether all frames are to be printed for each page. On the Macintosh, these options are divided between the Page Setup and the Print Margins dialog boxes.

The Print and Page Setup dialog boxes are standard in either operating system, and their appearance depends on the selected printer driver.

Set printing options and print frames

1 Select File > Page Setup (Windows) or File > Print Margins (Macintosh).

2 Set page margins. Select both Center options to print the frame in the center of the page.

3 In the Frames menu, select whether to print all frames in the document or only the first frame of each scene.

4 In the Layout menu, select from the following options:

Actual Size Prints the frame at full size. Enter a value for Scale to reduce or enlarge the printed frame.

Fit On One Page Reduces or enlarges each frame so it fills the print area of the page.

Storyboard Prints several thumbnails on one page. Select from Boxes, Grid, or Blank. Enter the number of thumbnails per page in the Frames box. Set the space between the thumbnails in the Story Margin box, and select Label Frames to print the frame label as a thumbnail.

5 To print frames, select File > Print.

Printing from SWF files at runtime using ActionScript 2.0

You can add printing functionality to Flash documents that lets users print from Flash Player. You can use the ActionScript™ PrintJob class, or you can use the print() or printAsBitmap() ActionScript functions. Users can also access the Flash Player context menu and select the Print command there. Additionally, users can print from a browser, rather than from Flash Player, by selecting a command such as File > Print from the browser window. However, printing from Flash Player directly, rather than from a browser window Print menu, offers several advantages, including the following:

- Users can print all frames or certain frames that you've labeled as printable from Flash Player. Additionally, you can set the print area of a frame.
- You can specify that content print as vector graphics (to take advantage of higher resolution) or as bitmaps (to preserve transparency and color effects).
- The ActionScript PrintJob object improves upon the print() and printAsBitmap() functions by adding the ability to print dynamically rendered pages as a single print job. The PrintJob object also provides the user's printer settings, which can be used to format reports specifically for the user.
- Flash Player versions earlier than 4.0.25 (Windows) or 4.0.20 (Macintosh) do not support printing frames directly. Flash Player 7 and later supports the PrintJob class.

***Note:** For information on printing from SWF files at runtime using ActionScript 3.0, see "Printing" in Programming ActionScript 3.0.*

Controlling printing

To control what users can print, remember the following items as you set up documents and movie clips for printing:

- Adjust the page layout in any frames that you designate as printable to match the desired printed output. Using Flash Player, you can print all shapes, symbols, bitmaps, text blocks, and text fields. Levels in a SWF file are not composited on print output.
- The printer driver uses the HTML settings for dimension, scale, and alignment in the Publish Settings dialog box. Use these settings to control the print layout.
- The selected frames print as they appear in the movie clip symbol. Let users print a movie clip that is not visible in a browser by using the Actions panel to set the movie clip's `_visible` property to `false`. Changing the property of a movie clip with the `setProperty` action, tweening, or any transformation tool does not affect how a movie clip prints.
- For a movie clip to be printable, it must be on the Stage or pasteboard and it must be given an instance name.
- All elements must be fully loaded to print. Use the movie clip `_framesloaded` property to check whether the printable content is loaded. For more information, see _framesloaded (MovieClip._framesloaded property) in the *ActionScript 2.0 Language Reference*.

***Note:** For information on printing from SWF files at runtime using ActionScript 3.0, see "Printing" in Programming ActionScript 3.0.*

Using the ActionScript 2.0 PrintJob class

The ActionScript PrintJob class, in addition to improving print functionality available with the `print()` method, lets you render dynamic content at runtime, prompt users with a single print dialog box, and print an unscaled document with proportions that map to the proportions of the content. This capability is useful for rendering and printing external dynamic content, such as database content and dynamic text.

Additionally, with properties populated by the `PrintJob.start()` method, your document can access your user's printer settings, such as page height, width, and orientation, and you can configure your document to dynamically format Flash content that is appropriate for the printer settings.

Build a print job

Because you are spooling a print job to the user's operating system between your calls to the `PrintJob.start()` and `PrintJob.send()` methods, and because the PrintJob functions might temporarily affect the Flash Player internal view of onscreen Flash content, implement print-specific activities only between your calls to `PrintJob.start()` and `PrintJob.send()`. For example, the Flash content should not interact with the user between `PrintJob.start()` and `PrintJob.send()`. Instead, complete formatting of your print job, add pages to the print job, and send the print job to the printer.

1. Create an instance of the print job object: `new PrintJob()`.
2. Start the print job and display the print dialog box for the operating system: `PrintJob.start()`.
3. Add pages to the print job (call once per page to add to the print job): `PrintJob.addPage()`.
4. Send the print job to the printer: `PrintJob.send()`.
5. Delete the print job: `delete PrintJob`.

The following example shows ActionScript code that creates a print job for a button:

```
myButton.onRelease = function()
{
    var my_pj = new PrintJob();
    var myResult = my_pj.start();
    if(myResult){
        myResult = my_pj.addPage (0, {xMin : 0, xMax: 400, yMin: 0,
            yMax: 400});
        myResult = my_pj.addPage ("myMovieClip", {xMin : 0, xMax: 400,
            yMin: 400, yMax: 800},{printAsBitmap:true}, 1);
        myResult = my_pj.addPage (1, null,{printAsBitmap:false}, 2);
        myResult = my_pj.addPage (0);
        my_pj.send();
    }
    delete my_pj;
}
```

Only one print job can run at any given time. A second print job cannot be created until one of the following events occurs with the previous print job:

- The print job is entirely successful and the `PrintJob.send()` method is called.
- The `PrintJob.start()` method returns a value of `false`.
- The `PrintJob.addPage()` method returns a value of `false`.
- The `delete PrintJob` method is called.

***Note:** For information on printing from SWF files at runtime using ActionScript 3.0, see "Printing" in Programming ActionScript 3.0.*

Starting a print job

Calling the ActionScript 2.0 `PrintJob.start()` method spools the print job to the user's operating system and prompts the user's operating system print dialog box to appear.

If the user selects an option to begin printing, the PrintJob.start() method returns a value of true. (The value is false if the user cancels the print job, in which case the script should call only delete). If successful, the PrintJob.start() method sets values for the paperHeight, paperWidth, pageHeight, pageWidth, and orientation properties.

Depending on the user's operating system, an additional dialog box might appear until spooling is complete and the PrintJob.send function is called: Make calls to PrintJob.addPage() and PrintJob.send() expeditiously. If ten seconds elapse between the PrintJob.start() function call and the PrintJob.send() function call, which sends the print job to the printer, Flash Player effectively calls PrintJob.send(), causing any pages that are added by using PrintJob.addPage() to be printed and spooling to stop.

When a new print job is constructed, the PrintJob() properties are initialized to 0. When PrintJob.start() is called, after the user selects the print option in the operating system print dialog box, Flash Player retrieves the print settings from the operating system. The PrintJob.start() function populates the following properties:

Property	Type	Unit	Notes
PrintJob.paperHeight	number	points	Overall paper height
PrintJob.paperWidth	number	points	Overall paper width
PrintJob.pageHeight	number	points	Height of actual printable area on the page; does not include any user-set margins
PrintJob.pageWidth	number	points	Width of actual printable area on the page; does not include any user-set margins
PrintJob.orientation	string	n/a	Portrait or landscape orientation

Note: *For information on printing from SWF files at runtime using ActionScript 3.0, see "Printing" in Programming ActionScript 3.0.*

Adding pages to a print job

You add pages to your print job with the ActionScript 2.0 PrintJob.addPage() method. Although the method can include as many as four parameters, the only required parameter is target/level. The three optional parameters are printArea, options, and frameNum.

If you are not using a particular optional parameter but are using other optional parameters, use NULL in place of the excluded optional parameter.

With all four parameters, the function uses the following syntax:

```
MyPrintJob.addPage(target[,printArea:Object, options:Object, frameNum:Number]):boolean;
```

If you provide an invalid parameter, the print job uses default parameter values.

Each call to add a new page is unique, which lets you modify parameters without affecting previously set parameters. For example, you can specify that one page print as a bitmap image and another page print as a vector graphic. You can add as many new pages to your print job as the print job requires. One call to add a page equals one printed page.

***Note:** Any ActionScript code that needs to be called to change a resulting printout must run before the `PrintJob.addPage()` method is called. The ActionScript can, however, run before or after a new `PrintJob()method`. If a frame has a call to `PrintJob.addPage()`, the call itself does not guarantee that the ActionScript script on that frame will run when that frame is printed.*

***Note:** For information on printing from SWF files at runtime using ActionScript 3.0, see "Printing" in Programming ActionScript 3.0.*

Specifying a target

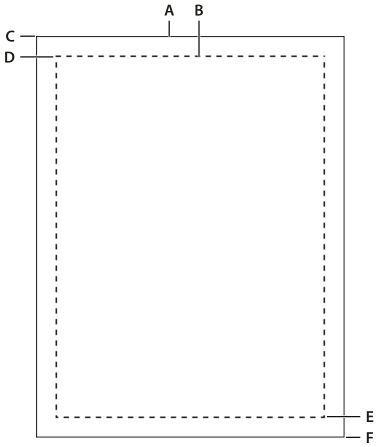
The ActionScript 2.0 `target` parameter can be either a number that represents a level (such as 0 for the _root document), or a string that represents the instance name of a movie clip (`"myMovieClip"`).

Specifying a print area

The optional `printArea` parameter includes the following values:

```
{xMin:Number, xMax:Number, yMin:Number, yMax:Number}
```

The `xMin`, `xMax`, `yMin`, and `yMax` values represent screen pixels relative to the target level or movie clip registration point. The print area orientation is from the upper-left corner of the printable area on the page. If the print area is larger than the printable area on the page, the print data that exceeds the right and bottom edge of the page is clipped.



***A.** Paper rectangle **B.** Page rectangle **C.** (594,774) **D.** (576,756) **E.** (0,0) **F.** (-18,-18)*

If you don't specify a print area, or if you specify an invalid print area, the print area defaults to the Stage area of the root document.

Scaling, points, and pixels

A print job using the PrintJob class prints Flash content, by default, without scaling it. For example, an object that is 144 pixels wide on screen prints as 144 points, or 2 inches wide.

One point equals one pixel. In the authoring tool, 72 pixels equals one inch; on paper, 72 points equals one inch.

To understand how Flash screen content maps to the printed page, it helps to understand screen and print units of measure. Pixels are a screen measurement and points are a print measurement. Both pixels and points equal 1/72 of an inch. A *twip* is 1/20 of a point and pixel.

The following list further illustrates the relationship between units of measure.

- 1 pixel = 20 twips
- 1 point = 20 twips
- 72 pixels = 1 inch
- 72 points = 1 inch
- 567 twips = 1 cm
- 1440 twips = 1 inch

To scale a movie clip before printing, set its `MovieClip.xscale` and `MovieClip.yscale` properties before calling this method, and set them back to their original values afterward. If you scale a movie clip and also pass a value for the *`printArea`* property, the pixel values passed to *`printArea`*reflect the original size of the movie clip. That is, if you set a movie clip's scale to 50% and specify a print area of 500 x 500 pixels, the content that prints is identical to the content that would print if you didn't scale the movie clip; however, it prints at half the size. For more information, see the addPage (PrintJob.addPage method)method of the `printjob` classin the *ActionScript 2.0 Language Reference.*

***Note:** For information on printing from SWF files at runtime using ActionScript 3.0, see "Printing" in Programming ActionScript 3.0.*

Specifying printing as a vector image or bitmap graphic

The `options` parameter lets you specify whether to print as a vector graphic or bitmap image. When using this optional parameter, use the following syntax:

```
{printAsBitmap:boolean}
```

The default value is `false`, which represents a request for vector printing. Remember the following suggestions when determining which value to use:

- If the content that you're printing includes a bitmap image, specify that the print job print as a bitmap to include any transparency and color effects.
- Conversely, if the content does not include bitmap images, specify that the print job print as vector graphics to take advantage of the higher image quality.

Specifying a frame to print

The `frameNum` parameter lets you specify a frame to print. If you do not specify a frame number parameter, by default the current frame of the target or level that you specify as the first parameter when you add a page prints.

***Note:** For information on printing from SWF files at runtime using ActionScript 3.0, see "Printing" in Programming ActionScript 3.0.*

Print or delete the print job

1 To send the print job to the printer after using the ActionScript 2.0 `addPage()` calls, use the `PrintJob.send()` method, which causes Flash Player to stop spooling the print job so that the printer starts printing.

2 To delete the print job after you send the print job to a printer, use the `delete PrintJob()` ActionScript function to delete the PrintJob object, which frees memory.

***Note:** For information on printing from SWF files at runtime using ActionScript 3.0, see "Printing" in Programming ActionScript 3.0.*

Printing frames independent of the PrintJob class

The ActionScript 2.0 PrintJob class, available for Flash Player 7 and later, offers many advantages over the `print()` and `printAsBitmap()` methods for printing. However, to print targeting Flash Player 6 and earlier versions, back to Flash Player 4.0.25 (Windows) and 4.0.20 (Macintosh), use `print()` and `printAsBitmap()` functions and frame labels.

To set up printing from Flash Player independent of the PrintJob class, specify frames to print and set their print area.

***Note:** For information on printing from SWF files at runtime using ActionScript 3.0, see "Printing" in Programming ActionScript 3.0.*

Specifying a print area when not using the PrintJob object

By default, when frames are printed, the document file's Stage determines the print area. Any object that extends off the Stage is clipped and does not print. Loaded movies use their own Stage size for the print area, not the main SWF file's Stage size.

As an alternative to using a document's Stage size, set the following print areas:

- For either the Flash Player context menu or the ActionScript 2.0 `print()` function, designate the bounding box for SWF content as the print area for all frames by selecting an object in one frame as the bounding box. This option is useful, for example, to print a full-page data sheet from a web banner.
- The `print()` function lets you use the composite bounding box of all printable frames in a Timeline as the print area—for example, to print multiple frames that share a registration point. To use the composite bounding box, use the `bMax` parameter, as shown in the following example:

```
print ("myMovie", "bmax")
```

- The `print()` function lets you change the print area for each frame, scaling objects to fit the print area—for example, to have objects of different sizes in each frame fill the printed page. To change the bounding box per frame, use the Frame parameter in the Print action parameters, as shown in the following example:

```
print ("myMovie", "bframe")
```

- The `print()` function lets you designate the bounding box of a specific frame in a document as the print area for all printable frames in the document, as shown in the following example:

```
print ("myMovie", "bmovie")
```

You use the frame label #b to designate a frame to be used to designate the print area. The label #b must be on the same layer as a frame labeled #p.

For more information about `print()` function parameters, see "print function" in the *ActionScript 2.0 Language Reference.*

***Note:** For information on printing from SWF files at runtime using ActionScript 3.0, see "Printing" in Programming ActionScript 3.0.*

Specify a print area when printing frames

1 Open the Flash document (FLA file) containing the frames you will set to print.

2 Select a frame that you have not specified to print with a #p frame label that is on the same layer as a frame labeled #p.

To organize your work, select the next frame after a frame labeled #p.

3 Create a shape on the Stage the size of the desired print area. To use a frame's bounding box, select a frame with any object of the appropriate print area size.

4 Select the frame in the Timeline that contains the shape to use for the bounding box.

5 In the Property inspector (Window > Properties > Properties), enter `#b` for Frame Label to specify the selected shape as the bounding box for the print area.

Only one #b frame label per Timeline is allowed. This option is the same as selecting the Movie bounding box option with the Print action.

Using the print() function when not using the PrintJob object

The basic syntax for the ActionScript 2.0 `print()` function, which you can associate with a button or other trigger in your document to activate printing, is as follows:

```
print (target, “Bounding box”);
```

The target parameter specifies the location of the frames that print, and the bounding box parameter specifies the print area.

Assign a `print()` function to a button, frame, or movie clip in your document to let users print Flash content. If you assign a `print()` function to a frame, the action executes when the playhead reaches the designated frame.

The `print()` function lets you print frames in other movie clips in addition to the main Timeline. Each `print()` function sets only one Timeline for printing, but the action lets you specify any number of frames in the Timeline to print. If you attach more than one `print()` function to a single button or frame, the Print dialog box appears for each action executed.

See also “print function” in the *ActionScript 2.0 Language Reference.*

Note: *For information on printing from SWF files at runtime using ActionScript 3.0, see “Printing” in Programming ActionScript 3.0.*

Change the printed background color

You can print the background color set in the Document Properties dialog box. Change the background color for only the frames to be printed by placing a colored object on the lowest layer of the Timeline being printed.

1 Place a filled shape that covers the Stage on the lowest layer of the Timeline that will print.

2 Select the shape and select Modify > Document. Select a color for the printing background.

This action changes the entire document’s background color, including that of movie clips and loaded SWF files.

3 Do one of the following:

- To print that color as the document’s background, designate to print the frame in which you placed the shape.
- To maintain a different background color for nonprinting frames, repeat steps 2 and 3. Then place the shape on the lowest layer of the Timeline, in all the frames that are not designated to print.

Use frame labels to disable printing

To choose not to print any of the frames in the main Timeline, label a frame as !#p to make the entire SWF file nonprintable. Labeling a frame as !#pdims the Print command in the Flash Player context menu. You can also remove the Flash Player context menu.

If you disable printing from Flash Player, the user can still use the browser Print command to print frames. Because this command is a browser feature, you cannot use Flash to control or disable it.

See also

“Publishing overview” on page 418

Disable printing in the Flash Player context menu

1 Open or make active the Flash document (FLA file) to publish.

2 Select the first keyframe in the main Timeline.

3 Select Window > Properties > Properties to view the Property inspector.

4 In the Property inspector, for Frame Label enter `!#p`to specify the frame as nonprinting.

Specify only one !#p label to dim the Print command in the context menu.

***Note:** You can also select a blank frame (rather than a keyframe) and label it* `#p`.

Disable printing by removing the Flash Player context menu

1 Open or make active the Flash document (FLA file) to publish.

2 Select File > Publish Settings.

3 Select the HTML tab and deselect Display Menu and click OK.

Print from the Flash Player context menu

Use the Print command in the Flash Player context menu to print frames from any Flash SWF file.

The context menu's Print command cannot print transparency or color effects and cannot print frames from other movie clips; for more advanced printing capabilities, use the PrintJob object or the `print()` function.

1 Open the document.

The command prints the frames labeled #p by using the Stage for the print area or the specified bounding box.

If you haven't designated specific frames to print, all frames in the document main Timeline print.

2 Select File > Publish Preview > Default or press F12 to view your Flash content in a browser.

3 Right-click (Windows) or Control-click (Macintosh) in the Flash content in the browser window to display the Flash Player context menu.

4 Select Print from the Flash Player context menu to display the Print dialog box.

5 In Windows, select the print range to select which frames to print.

6 On the Macintosh, in the Print dialog box, select the pages to print.

7 Select other print options, according to your printer's properties.

8 Click OK (Windows) or Print (Macintosh).

***Note:** Printing from the context menu does not interact with calls to the* `PrintJob` *object.*

Publishing a document with printable frames

Publish a Flash document with printable frames to the web by using the Publish command to generate the necessary Flash HTML templates.

Users must have Flash Player 4.0.25 (Windows) or 4.0.20 (Macintosh) or later to take advantage of any print functionality you added and to print the designated frames in Flash. Set up a detection scheme to check for the proper Flash Player version.

***Note:** When you use the PrintJob class, users must have Flash Player 7 or later.*

See also

"Publishing overview" on page 418

Chapter 23: Best practices

Adobe® Flash® CS3 Professional contains features and capabilities that make it a flexible tool and allow more than one way to do the same thing in Flash. Over time, the Flash community has developed preferred methods for accomplishing many common tasks.

Best practices encourage consistency when you work on Flash or ActionScript™ documents, share FLA or ActionScript files, work on applications, and when you are learning or teaching Flash and ActionScript.

Structuring FLA files

Organizing timelines and the library

Frames and layers on a timeline show you where assets are placed and determine how your document works. How a timeline and the library are set up and used affect the entire FLA file and its overall usability. The following guidelines help you author content efficiently, and let other authors who use your FLA documents have a greater understanding of how the document is structured.

- Give each layer an intuitive layer name, and place related assets together in the same location. Avoid using the default layer names (such as Layer 1, Layer 2).

Clearly describe the purpose or content of each layer or folder when you name them.

If applicable, place your layers that include ActionScript and a layer for frame labels at the top of the layer stack in the timeline. For example, name the layer that contains your ActionScript actions.

- Use layer folders to group and organize similar layers, to facilitate locating the layers that include code and labels.
- Lock layers that you are not using or do not want to modify. Lock your ActionScript layer immediately so that symbol instances or media assets are not placed on that layer.
- Never put any instances or assets on a layer that includes ActionScript. Because this can potentially cause conflicts between assets on the Stage and ActionScript that references them, keep all of your code on its own actions layer, and lock it after you create it.
- Use frame labels in a FLA file instead of using frame numbers in your ActionScript code if you reference frames in your code. If those frames change later when you edit the timeline, and you use frame labels and move them on the timeline, you do not have to change any references in your code.
- Use library folders.

Use folders in the library to organize similar elements (such as symbols and media assets) in a FLA file. If you name library folders consistently each time you create a file, it is easier to remember where you put assets. Commonly used folder names are Buttons, MovieClips, Graphics, Assets, Components, and, sometimes, Classes.

Using scenes

Using scenes is similar to using several SWF files to create a larger presentation. Each scene has a timeline. When the playhead reaches the final frame of a scene, the playhead progresses to the next scene. When you publish a SWF file, the timeline of each scene combines into a single timeline in the SWF file. After the SWF file compiles, it behaves as if you created the FLA file using one scene. Because of this behavior, avoid using scenes for the following reasons:

- Scenes can make documents confusing to edit, particularly in multiauthor environments. Anyone using the FLA document might have to search several scenes within a FLA file to locate code and assets. Consider loading content or using movie clips instead.
- Scenes often result in large SWF files.
- Scenes force users to progressively download the entire SWF file, instead of loading the assets they actually want to see or use. If you avoid scenes, the user can control what content they download as they progress through your SWF file. The user has more control over how much content they download, which is better for bandwidth management. One drawback is the requirement for managing a greater number of FLA documents.
- Scenes combined with ActionScript might produce unexpected results. Because each scene timeline is compressed onto a single timeline, you might encounter errors involving your ActionScript and scenes, which typically requires extra, complicated debugging.

If you create lengthy animations, you might find it advantageous to use scenes. If disadvantages apply to your document, consider using multiple FLA files, or movie clips to build an animation instead of using scenes.

Saving files and version control

When you save your FLA files, use a consistent naming scheme for your documents. This is particularly important if you save multiple versions of a single project.

Some problems might occur if you only work with one FLA file and do not save versions when you create the file. Files might become larger because of the history that's saved in the FLA file, or become corrupt (as with any software you use) while you are working on the file.

If you save multiple versions while developing, you have an earlier version available if you need to revert.

Use intuitive names for your files that are easy to read, not cryptic, and work well online:

- Do not use spaces, capitalization, or special characters.
- Only use letters, numbers, dashes, and underscores.
- If you save multiple versions of the same file, devise a consistent numbering system such as menu01.swf, menu02.swf and so on.
- Consider using all lowercase characters in your naming schemes, because some server software is case sensitive.
- Consider a naming system that uses a noun-verb or adjective-noun combination for naming files, for instance, classplanning.swf and myproject.swf.

Use the following methods to save new versions of a FLA file when you build an extensive project:

- Select File > Save As, and save a new version of your document.
- Use version control software or the Project panel to control your Flash documents.

Note: *SourceSafe on Windows is the only officially supported version control software that integrates with the Project panel. You can use other version control software packages with FLA documents, but not necessarily in the Project panel.*

You can use several options to save a file: Save, Save As, and Save and Compact. When you save a file, Flash does not analyze all the data before creating an optimized version of the document. Instead, the changes you make to the document are appended to the end of the FLA file's data, which shortens the time it takes to save the document. When you Save As, Flash writes a new and optimized version of the file, which results in a smaller file size. When you Save and Compact, Flash creates a new optimized file (removing the undo history) and deletes the original file.

Important: *If you select Save when working with a document, you can undo before that save point. Because Save and Compact deletes the earlier version of the file and replaces it with the optimized version, you cannot undo earlier changes.*

If you are not using version control software to create backups of your FLA file, use Save As and type a new file name for your document after every milestone in your project.

Many software packages allow users to use version control with their files, which enables teams to work efficiently and reduce errors (such as overwriting files or working on old versions of a document). As with other documents, you can use these programs to organize the Flash documents outside Flash.

See also

"Adobe Version Cue" on page 82

"Projects and version control guidelines" on page 472

Organizing ActionScript in an application

Keeping actions together

Whenever possible, put your ActionScript in a single location. Organizing your code in one place helps you edit projects more efficiently, because you can avoid searching in different places when you debug or modify the ActionScript. If you put code in a FLA file, put ActionScript on Frame 1 or Frame 2 in a layer called *actions* on the topmost layer in the Timeline. Alternatively, you might put all of your code in ActionScript files. Some Flash applications do not always put all code in a single place (in particular, ActionScript 2.0-based applications that use screens or behaviors).

You can usually put all your code in the same location (on a frame, or in ActionScript files), with the following advantages:

- Code is easy to find in a potentially complex source file.
- Code is easy to debug.

See also

"Attaching code to objects" on page 467

"Behaviors conventions" on page 467

"Using the MVC design pattern" on page 476

"Organizing files and storing code" on page 476

Attaching code to objects

Avoid attaching ActionScript to objects in a FLA file, even in simple SWF files. (Only ActionScript 1.0 and 2.0. can be attached to objects; ActionScript 3.0 cannot.) Attaching code to an object means that you select a movie clip, component, or button instance; open the Actions panel; and add ActionScript using the `on()` or `onClipEvent()` handler functions.

Attaching ActionScript code to objects is strongly discouraged for the following reasons:

- It is difficult to locate, and the FLA files are difficult to edit.
- It is difficult to debug.
- ActionScript that is written on the timeline or in classes is more elegant and easier to build upon.
- It encourages poor coding style.
- The contrast between two styles of coding can be confusing to people learning ActionScript; it forces students and readers to learn different coding styles, additional syntax, and a poor and limited coding style.

Avoid attaching ActionScript 2.0 to a button called `myButton_btn`, which looks like the following:

```
on (release) {
    //do something
}
```

However, placing ActionScript 2.0 with the same purpose on the timeline (which is encouraged), looks like the following code:

```
myButton_btn.onRelease = function() {
    //do something
};
```

***Note:** Different practices apply when using behaviors, which sometimes involves attaching code to objects.*

See also

“Comparing timeline code with object code” on page 468

Behaviors conventions

About behaviors conventions

Behaviors are prewritten ActionScript 2.0 code snippets that you can add to parts of a FLA file. Many developers enter ActionScript code either into one or several frames on the main Timeline or in external ActionScript files. However, when you use behaviors, sometimes code is placed directly on symbol instances (such as buttons, movie clips, or components) instead of being placed on the timeline.

Behaviors are not supported by ActionScript 3.0.

Comparing timeline code with object code

To avoid problems that decentralized ActionScript 2.0 code creates, carefully plan a document that uses behaviors. Many developers do not place ActionScript on symbol instances, and instead place their code on the Timeline (timeline code) or in classes. Because behaviors add code to many locations in a FLA file, your ActionScript is not centralized and can be difficult to locate. When code is not centralized, it is difficult to understand interactions between the snippets of code, and it is impossible to write elegant code. Decentralized code can potentially lead to problems debugging code or editing files.

If you use behaviors, try the following features to facilitate working with behaviors and decentralized ActionScript:

Script Navigator Makes your timeline code or code on individual objects easy to find and edit in the Actions panel.

Find And Replace Lets you search for strings and replace them in a FLA file.

Script Pinning Lets you pin multiple scripts from various objects and work with them simultaneously in the Actions panel. This method works best with the Script navigator.

Movie Explorer Lets you view and organize the contents of a FLA file, and select elements (including scripts) for further modification.

When to use behaviors

The main difference between a FLA file with behaviors and a FLA file without behaviors is the workflow you must use for editing the project. If you use behaviors, you must select each instance on the Stage, or select the Stage, and open the Actions or Behaviors panel to make modifications. If you write your own ActionScript and put all your code on the main Timeline, you only have to make your changes on the Timeline.

If you have a FLA file with symbols, you can select one of the instances on the Stage, and use the Add menu on the Behaviors panel to add a behavior to that instance. The behavior you select automatically adds code that attaches to the instance, using "object code" such as the `on()` handler. You can also select a frame on a timeline, and add different behaviors to a frame using the Behaviors panel.

Decide how to structure your FLA file. Examine how and where to use behaviors and ActionScript in your FLA file. Consider the following questions:

- What code do the behaviors contain?
- Do you have to modify the behavior code? If so, by how much? To modify the behavior code to any extent, do not use behaviors. You usually cannot edit behaviors by using the Behaviors panel if you make modifications to the ActionScript. To significantly edit the behaviors in the Actions panel, it is usually easier to write all of the ActionScript yourself in a centralized location.
- What other ActionScript do you need, and does other ActionScript have to interact with the behavior code? Debugging and modifications are easier to make from a central location. For example, if code on a timeline interacts with behaviors placed on objects, avoid behaviors.
- How many behaviors do you have to use, and where do you plan to put them in the FLA file? If your behaviors are all placed on a timeline, they might work well in your document. Or, your workflow might not be affected if you use only a small number of behaviors. However, if you use many behaviors on a lot of object instances, writing your own code on the Timeline or in external ActionScript files might be more efficient.

Remember, ActionScript 3.0 does not support behaviors.

Using behaviors consistently

Use behaviors consistently throughout a document when they are your main or only source of ActionScript. Use behaviors when you have little or no additional code in the FLA file, or have a consistent system in place for managing the behaviors that you use.

If you add ActionScript to a FLA file, put code in the same locations where behaviors are added, and document how and where you add code.

For example, if you place code on instances on the Stage (object code), on the main Timeline (frame scripts), and also in external AS files, examine your file structure. Your project will be difficult to manage if you have code in all of these places. However, if you logically use behaviors and structure your code to work in a particular way surrounding those behaviors (place everything on object instances), at least your workflow is consistent. The document will be easier to modify later.

Sharing files that use behaviors

If you plan to share your FLA file with other users and you use ActionScript placed on or inside objects (such as movie clips), it can be difficult for those users to find your code's location, even when they use the Movie Explorer to search through the document.

Document the use of behaviors if you are working with a complex document. Depending on the size of the application, create a flow chart, list, or use good documentation comments in a central location on the main Timeline.

If you are creating a FLA file with code placed in many locations throughout the document and plan to share the file, leave a comment on Frame 1 on the main Timeline to tell users where to find the code and how the file is structured. The following example shows a comment (on Frame 1) that tells users the location of the ActionScript:

```
/*
    ActionScript placed on component instances and inside movie clips using behaviors.
    Use the Movie Explorer to locate ActionScript
*/
```

***Note:** This technique is not necessary if your code is easy to find, the document is not shared, or all of your code is placed on frames of the main Timeline.*

Video conventions

About video conventions

Many options are available to edit video before you import it into a FLA document, or load an FLV file into a SWF file. Flash and Flash Video Encoder have greater controls for video compression. Compressing video carefully is important because it controls the quality of the video footage and the size of the file. Video files, even when compressed, are large in comparison with most other assets in your SWF file.

***Note:** Provide the user with control over the media in a SWF file. For example, if you add audio to a document with video (or even a looping background sound), let the user control the sound.*

Using video in an application

Before you import video into Flash, consider what video quality you need, what video format to use with the FLA file, and how to download it. When you import video into a FLA file (called *embedded video*), it increases the size of the SWF file that you publish. This video starts progressively downloading to the user's computer whether or not they view the video.

You can also progressively download or stream the video at runtime from an external Flash Video (FLV) file on your server. When it starts downloading depends on how you structure your application.

***Note:** Video progressively downloads from the server like SWF files, which is not actually streaming. Dynamically loading content has distinct advantages over keeping all your content in a single SWF file. For example, you will have smaller files and quicker loading, and the user only downloads what they want to see or use in your application.*

You can display external FLV video using a component or a video object. A component makes developing applications with FLV video easy, because the video controls are prebuilt, and you only need to specify an FLV file path to play the content. To keep your SWF file as small as possible, display video in a video object and create your own assets and code to control the video. Also consider using the FLVPlayback component in Adobe® Flash® CS3 Professional, which has a smaller file size than Media components (Flash MX Professional 2004 and later).

It is a good idea to give users some control (such as the ability to stop, pause, play, and resume the video, and control volume) over the video in a SWF file.

To gain certain kinds of flexibility over your video, such as manipulating the video with animation, or syncing various parts of it with the timeline, embed the video in the SWF file rather than loading it using ActionScript or one of the Media components.

For more control over a video instance than the Video class allows, place video inside a movie clip instance. The video's timeline plays independently from a Flash timeline, and you can place the content inside a movie clip to control timelines. You do not have to extend your main Timeline by many frames to accommodate for the video, which can make working with your FLA file difficult.

See also

"Working with video" on page 300

Exporting, compressing, and hosting FLV files

You can export FLV files from Adobe® Flash® CS3 Professional authoring environments. After you import video into your document, it appears as a video symbol in the library. To export the video as FLV, select the video and then select Library > Properties. To make export settings, click Export in the Embedded Video Properties dialog box.

The FLV QuickTime Exporter compresses video from third-party video editing software. You can also use Flash 8 Video Encoder, which is a stand-alone application that you can use to create FLV files. Because you have a lot of control over the export settings, the quality of the FLV file that is created using either of these tools is better than video exported directly from Flash.

When you compress video, remember the following recommendations:

- Do not recompress video.

Recompressing video leads to quality degradation, such as artifacts. Use raw footage or the least compressed footage that is available to you.

- Make your video as short as possible.

Trim the beginning and end of your video, and edit your video to remove any unnecessary content. This can be accomplished directly in Flash using the Video Import wizard.

- Adjust your compression settings.

If you compress footage and it looks great, try changing your settings to reduce the file size. Test your footage, and modify it until you find the best setting possible for the video you are compressing. All video has varying attributes that affect compression and file size; each video needs its own setting for the best results.

- Limit effects and rapid movement.

Limit movement if you are concerned about file size. Any kind of movement, particularly with many colors, increases file size. For example, effects (such as cross fades, blurs, and so on) increase file size, because the video contains more information.

- Choose appropriate dimensions.

If your target audience has a slow Internet connection (such as phone modems), make the dimensions of your video smaller, such as 160x120 pixels. If your visitors have fast connections, you can make your dimensions larger (for example, 320x240 pixels).

- Choose appropriate frames per second (fps).

If you target users that typically have older computer processors, choose a low rate of frames per second (such as 7 or 15 fps). If you target users with newer computers, you can use a higher rate of frames per second (such as 15 or 30 fps). Always choose an fps that is a multiple of your original frame rate. For example, if your original frame rate was 30 fps, compress to 15 fps or 7.5 fps.

- Choose an appropriate number of keyframes.

Video keyframes are different from keyframes in Flash. Each keyframe is a frame that draws when the video is compressed, so the more frequent your keyframes are the better quality the footage is. More keyframes also mean a larger file size. If you choose 30, a video keyframe draws every 30 frames. If you choose 15, the quality is higher because a keyframe draws ever 15 frames and the pixels in your footage are more accurate to the original.

- Reduce noise.

Noise (scattered pixels in your footage) increases file size. Reduce noise using your video editor, to reduce the video file size. Using more solid colors in your video reduces its file size.

When you put FLV files on a server, they use the `video/x-flv FLV mime type`. If you have difficulty viewing FLV files after you upload your files, check that this mime type is set on your server. FLV files are binary, and some applications that you build might require that the `application/octet-stream` subtype is also set. For more information on the Flash Player specifications, see www.adobe.com/go/flashfileformat.

Troubleshooting video

You can create an application and then encounter problems after you upload it to your server.

- Check that your Flash Player version is correct.

For example, if you encoded your files using On2 codec, you need Flash Player 8 or later installed for the browsers you use to view your Flash content.

***Note:** For Flash Player and FLV compatibility, see About using FLV video in Learning ActionScript 2.0 in Adobe Flash.*

- Check that your server supports the FLV mime type. For more information on FLV files on a server, see Configuring your server for FLV files in *Learning ActionScript 2.0 in Adobe Flash*.

- Check security guidelines.

If you load FLV files from another server, make sure that you have the proper files or code in place to load from that external server. For information on policy files, see Server-side policy files for permitting access to data in *Learning ActionScript 2.0 in Adobe Flash*. For information about loading and security, see Understanding security in *Learning ActionScript 2.0 in Adobe Flash*.

- Check your target paths to your video are correct. If you use relative paths (such as /video/water.flv), try using absolute paths (such as http://www.helpexamples.com/flash/video/water.flv). If your application doesn't work as a relative path, but does work as an absolute path, correct the relative path.

Projects and version control guidelines

About projects and version control

Projects in Flash let members on a team work together on a single Flash application or project. A project file remembers each of the files it contains, and lets you incorporate some SourceSafe capabilities into your applications, which helps you keep backups of modified files.

Note: *Adobe® Flash® CS3 Professional (and earlier) does not support Microsoft Visual SourceSafe for version control on the Macintosh.*

To group multiple files into a single project file, use the Project panel. This helps simplify application building, where managing related files could get complex and confusing. You can define a site for your work, create a Flash project (FLP) file, and then upload everything to the server so that a team can work on the project.

Version control lets you check files in and out of your repository, and check that only one person is working on a file at a certain time. Other benefits include the ability to revert to older versions of the files.

Version control features help you ensure that you use the correct current files when authoring, and that certain files are not overwritten.

You can typically use your current source control software with Flash, but you might not be able to integrate it with the Project panel. Microsoft Visual SourceSafe is currently supported. Other software programs can manage and control your Flash documents, but you probably cannot integrate them with the Project panel.

See also

"Create and manage projects" on page 60

Administering projects

Assign an administrator to the project. This individual is responsible for creating and maintaining the project's structure. For example, documents are divided logically using folders to combine similar files. Typically, several authors work on one Flash project.

Important: *The administrator is the only person who changes the project file and structure.*

The administrator performs the following tasks:

- Confirms changes that are made to the project's structure, which encourages project stability.
- Defines the site, and creates the Flash project (FLP), main FLA document, and any subdirectories for the project's assets. These directories might include media, images, or classes that dynamically load into the project.

- Uploads everything to the server.
- Creates a clear structure for the project, and communicates how it works and where to add additional assets (such as class and image files) to everyone who is working on the application.

Authoring projects

Authors on a Flash project do not change the project root, directory structure of the project, or the site. This includes adding, removing, or changing subdirectory names, or adding additional subdirectories to the project on their local computer. If individual authors change the site or project structure, the local files are out of sync with those on the server. This causes problems in the application, such as class path and missing file errors, and so on. Individual authors can copy assets to the subdirectory files that the project's administrator creates.

- Each author on a Flash project selects File > Open from Site, selects the name of the site, and then selects the project's FLP file.
- The author updates the project with any missing files. This ensures that the author is working with the latest version of the site.
- When the author selects Yes, all the project files download to the author's local computer, so the structure on the local computer matches the structure on the server.

Changing structure

When the project's structure needs to be changed, authors check in all their files. The project's administrator checks out all the files to make any necessary changes. After this is done, each person working on the project deletes the root folder of their own local copy of the project. To avoid accidentally using legacy files, each author uses File > Open from Site to download a new copy of the site. This process helps prevent accidentally using legacy files, and reduces similar versioning problems.

Flash application authoring guidelines

About Flash application guidelines

The best way to create Flash applications depends on the application you create and the technology that you are using to build the application.

An online application lets a user influence a website by interacting with it. For example, the application might collect information from the user (such as a username and password for a registration), information might be added to the site (such as in a forum), or the user might interact in real time with other site visitors (such as a chat room or interactive white board). Results from the server often appear in the SWF file, depending on the interaction. These examples are applications that involve the user and different kinds of server interaction. A website that does not use visitor information or data is not an application (for example, a portfolio, cartoon animation, or static informational site). Flash applications involve an interactive process between the user, a web application, and a server. The basic process is as follows:

1. A user enters information into a SWF file.
2. The information is converted into data.
3. The data is formatted and sent to a web server.
4. The data is collected by the web server and sent to an application server (for example, ColdFusion, PHP, or ASP).

5 The data is processed and sent back to the web server.

6 The web server sends the results to the SWF file.

7 The SWF file receives the formatted data.

8 Your ActionScript processes the data so the application can use it.

When you build an application, you must select a protocol for transferring data. The protocol alerts the application when data is sent or received, in what format the data is transferred, and how it handles a server's response. After data is received in the SWF file, it must be manipulated and formatted. If you use a protocol, you do not have to worry about data being in an unexpected format. When you transfer data using name-value pairs, you can check how the data is formatted. Check that the data is formatted correctly, so you do not receive XML formatted data and so the SWF file knows what data to expect and work with.

Collecting and formatting data

Applications depend on user interaction with the SWF file. Frequently, it depends on the user entering data into forms. Flash provides many ways you can enter and format data in Flash applications. This flexibility exists because of the capabilities you have with animation and creative control over the interface, and error checking and validation you can perform using ActionScript.

Benefits from using Flash to build forms to collect data include the following:

- Increased design control.
- Decreased or no need for page refreshing.
- Reuse of common assets.

To save information that you collect from the user, save it in a shared object on the user's computer. Shared objects let you store data on a user's computer, which is similar to using a cookie. For more information on Shared objects, see the sharedObject class in ActionScript 2.0 Language Reference or ActionScript 3.0 Language and Components Reference.

Sending and processing data

You must typically process information before you send it to the server, so it's formatted in a way that the server understands. When the server receives the data, it can be manipulated in any number of ways and sent back to the SWF file in a format that it can accept, which can range from name-value pairs to complex objects.

Note: *Your application server must have the MIME type of its output set to* `application/x-www-urlform-encoded`*. If that MIME type is missing, the result is usually unusable when it reaches Flash.*

The following table shows you several options for sending data to a server and receiving data using Flash:

Send data	Description
LoadVars.send and LoadVars.sendAndLoad	Sends name-value pairs to a server-side script for processing. LoadVars.send sends variables to a remote script and ignores any response. LoadVar.sendAndLoad sends name-value pairs to a server and loads or parses the response into a target LoadVars object.
XML.send and XML.sendAndLoad	Similar to LoadVars, but XML.send and XML.sendAndLoad send XML packets instead of name-value pairs.
getURL	Using the getURL() function or MovieClip.getURL method, you can send variables from Flash to a frame or pop-up window.
Flash Remoting	Lets you easily exchange complex information between Flash and ColdFusion, ASP.NET, Java, and more. You can also use Flash Remoting to consume web services.
Web services	Adobe® Flash® CS3 Professional includes the WebServiceConnector component that lets you connect to remote web services, send and receive data, and bind results to components. This lets Flash developers quickly create Rich Internet Applications without having to write a single line of ActionScript. You can consume remote web services by using WebServiceClasses, which can require writing complex ActionScript.

Adding data loading and validation

Validate any information you retrieve before you send that data to a server. This reduces strain on the remote server, because it does not handle as many requests when users do not fill in required fields. Never rely solely on client-side validation in any application; server-side validation must also occur.

Even if you build a simple registration or login form, check that the user has entered their name and password. Perform this validation before sending the request to the remote server-side script and waiting for a result. Do not rely only on server-side validation. If a user enters only a username, the server-side script must receive the request, validate the data being sent, and return an error message to the Flash application, stating that it requires both the username and password. Likewise, if validation is performed only on the client side (within the SWF file), a user might hack the SWF file, bypass the validation, and send data to your server in an attempt to post the bad data.

Client-side validation can be as simple as making sure that a form field is at least one character long, or that the user entered a numeric value and not a string. To validate an e-mail address, for example, check that the text field in Flash isn't empty and contains at least the at sign (@) and dot (.) characters. For the server-side validation, add more complex validation and check that the e-mail address belongs to a valid domain.

You must write ActionScript to handle the data that loads into the SWF file from the server. After you finish loading data into a SWF file, the data can be accessed from that location. Use ActionScript to check whether the data is fully loaded. You can use callback functions or listeners to send a signal that the data is loaded into the document.

When you load data, it can be formatted in several ways:

- You might load XML, in which case you use the XML class methods and properties to parse the data and use it. If you use name-value pairs, the pairs turn into variables and you can manipulate them as variables.
- You might receive data from a web service or from Flash Remoting.

In both cases, you could receive complex data structures, such as arrays, objects, or record sets, which you must parse and bind appropriately.

Using error handling and debugging

Your application needs to be robust enough to anticipate certain errors and handle them accordingly.

One of the best ways to perform error handling in ActionScript 2.0 is to use the `try-catch-finally` blocks that let you throw and catch custom errors. By creating custom error classes, you can reuse code throughout your application without having to rewrite error handling code. For more information on throwing custom errors, see the `Error` class in *ActionScript 2.0 Language Reference*. For more information on `try-catch-finally` blocks, see `try..catch..finally` in *ActionScript 2.0 Language Reference*.

In ActionScript 3.0, use the `flash.errors` class to catch errors.

For more information, see “Handling synchronous errors in an application” in *Programming ActionScript 3.0*.

Organizing files and storing code

Consider the following guidelines before you start organizing files and storing code:

- Do you divide the SWF file into multiple SWF files, and, if so, how should they interact?
- What assets can you share across SWF files?
- What files do you dynamically load?
- How and where do you store ActionScript?

When you develop an application, store your server-side code and files in a logical directory structure, similar to those in an ActionScript package. Arrange your code this way to keep it well organized and reduce the risk of the code being overwritten.

For larger applications, encapsulate client-server communication and services in classes. When you use classes, you benefit in the following ways:

- You can reuse the code in more than one SWF file.
- You can edit code in a central place, and update all SWF files by republishing them.
- You can create a single API that can manipulate different UI elements or other assets that perform similar functions.

Using the MVC design pattern

The MVC design pattern is used to separate the information, output, and data processing in the application. The application is divided into three elements: model, view, and controller; each element handles a different part of the process.

The model Incorporates the data and rules of the application. Much of the application’s processing occurs in this part of the design pattern. The model also contains any components (such as CFCs, EJBs, and web services), and the database. Data returned is not formatted for the interface (or front end) of the application in this part of the process. The returned data can be used for different interfaces (or views).

The view Handles the front end of the application (the interface with which the user interacts), and renders the model’s contents. The interface specifies how the model’s data is presented and outputs the view for the user to use, and lets the user access or manipulate the application’s data. If the model changes, the view updates to reflect those changes by either pushing or pulling data (sending or requesting data). If you create a hybrid web application (for example, one that includes Flash interacting with other applications on the page), consider the multiple interfaces as part of the view in the design pattern. The MVC design pattern supports handling a variety of views.

The controller Handles the requirements of the model and view to process and display data, and typically contains a lot of code. It calls any part of the model, depending on user requests from the interface (or view), and contains code that's specific to the application. Because this code is specific to the application, it is usually not reusable. However, the other components in the design pattern are reusable. The controller does not process or output any data, but it takes the request from the user and decides what part of the model or view components it needs to call, and determines where to send the data and what formatting is applied to the returned data. The controller ensures that views have access to parts of the model data that they must display. The controller typically transmits and responds to changes that involve the model and view.

Each part of the model is built as a self-contained component in the overall process. If you change one part of the model (for example, you might rework the interface), the other parts of the process do not usually need modification, which reduces problems. If your design pattern is created correctly, you can change the view without reworking the model or controller. If your application does not use MVC, making changes anywhere can cause a rippling effect across all your code, which requires many more changes than if you were using a specific design pattern.

An important reason to use the MVC pattern is to separate data and logic from the user interface. By separating these parts of the process, you can have several different graphical interfaces that use the same model and unformatted data. This means that you can use your application with different Flash interfaces, such as an interface for the web, one for Pocket PC, a version for cell phones, and perhaps an HTML version that doesn't use Flash at all. Separating data from the rest of the application can greatly reduce the time it takes to develop, test, and even update more than one client interface. Similarly, adding new front ends for the same application is easier if you have an existing model to use.

Only use MVC if you build a large or complex application, such as an e-commerce website or an e-learning application. Using the architecture requires planning and understanding how Flash and this design pattern work. Carefully consider how the different pieces interact with each other; this typically involves testing and debugging. When you use MVC, testing and debugging are more involved and difficult than in typical Flash applications. If you build an application in which you need the additional complexity, consider using MVC to organize your work.

Creating secure applications

Dishonest users might try to hack your application, whether you build a small portal site where users can log in and read articles or a large e-commerce store. For this reason, consider the following steps to secure your application.

- Post data to HTTPS for data that needs to be secured. Encrypt values in Flash before sending them to a remote server to be processed.

Important: *Never store any information or code in a SWF file that you don't want users to see. It is easy to disassemble SWF files and view their contents using third-party software.*

- Add a cross-domain policy, which prevents unauthorized domains from accessing your assets.

Accessibility guidelines

About accessibility guidelines

Screen readers are complex, and you can easily encounter unexpected results in FLA files developed for use with screen readers, which is software that visually impaired users run to read websites aloud. Text is read aloud using specially designed software. A screen reader can only interpret textual content. However, any descriptions that you provide for the overall SWF file, movie clips, images, or other graphical content are also read aloud. Write descriptions for the important images and animations so that the screen reader can also interpret these assets in your SWF file. This is the SWF file equivalent to *alt* text in an HTML web page.

***Note:** Flash applications must be viewed in Internet Explorer on Windows, because Microsoft Active Accessibility (MSAA) support is limited to this browser.*

Flash Player uses Microsoft Active Accessibility (MSAA) to expose Flash content to screen readers. MSAA is a Windows-based technology that provides a standardized platform for information exchange between assistive technologies, such as screen readers, and other applications. Events (such as a change in the application) and objects are visible to screen readers by using MSAA.

***Note:** Flash Player 7 (and later) does not work with all screen-reader technologies. The third-party software provider must handle the information that MSAA provides.*

Creating accessible sites

Making a website accessible involves several different criteria:

Expose the information to screen readers **Make text or images realizable** Some visitors might have difficulty reading small text or seeing small graphics. Allow users to zoom in on these elements, taking advantage of scalable vector graphics in SWF files.

Provide audio narration Consider providing an audio narration for visitors without a screen reader, or where screen readers might not work, such as with video content.

Provide captions for audio narrations Some visitors might not be able to hear an audio narration for your site or a video. Consider providing captions for these visitors.

Do not rely on color to communicate information Many visitors might be color blind. If you rely on color to communicate information (such as: Click the green button to go to page 1, click the red button to go to page 2), provide text or speech equivalents.

Historically, many online presentations (such as videos) provide alternative ways for visually impaired visitors to access the content, for instance, a textual description of a video. However, Flash provides textual information directly to the screen reader. Although this usually means you need to make additional settings or ActionScript in a FLA file, you do not have to create a completely separate version.

Parts of your SWF file can be exposed to screen readers. Text elements (such as text fields, static text, and dynamic text), buttons, movie clips, components, and the entire SWF file can be interpreted by MSA-compliant screen readers.

Section 508 is United States legislation that provides guidelines for making information accessible to people with disabilities. Section 508 specifically addresses the need for websites to be accessible in several ways. Some websites, including all federal websites, must comply with these guidelines. If a SWF file does not communicate all of the information to the screen reader, the SWF file is no longer Section 508-compliant. For more information, see the Section 508 website.

Many nations have specified guidelines to follow to create accessible web sites, or follow guidelines established by other organizations. For more information on accessibility and web standards, see the World Wide Web Consortium (W3C) Web Accessibility Initiative website. These standards and guidelines describe what factors you must address when you create accessible HTML websites, and some of this information applies to Flash.

Exposing SWF file structure and navigation

Because of the visual nature of some SWF files, the layout and navigation of the page can be complex and difficult for screen readers to translate. An overall description of the SWF file is important to communicate information about its structure and how to navigate through the site's structure. You can provide this description by clicking the Stage and entering a description into the Accessibility panel. You can also create a separate area of the site to provide this description or overview.

Note: *If you enter a description for the main SWF file, this description is read each time the SWF file refreshes. You can avoid this redundancy by creating a separate informational page.*

Inform the user about any navigational elements that change in the SWF file. Perhaps an extra button is added, or the text on the face of a button changes, and this change is read aloud by the screen reader. Flash Player 7 and later supports updating these properties by using ActionScript. You can update the accessibility information in your applications if the content changes at runtime.

See also

"Creating accessibility with ActionScript" on page 362

Controlling descriptions and repetition

Designers and developers can assign descriptions for the animations, images, and graphics in a SWF file. Provide names for graphics so the screen reader can interpret them. If a graphic or animation does not communicate vital information to the SWF file (perhaps it is decorative or repetitive), or you outlined the element in the overall SWF file description, do not provide a separate description for that element. Providing unnecessary descriptions can be confusing to users who use screen readers.

Note: *If you divide text or use images for text in your SWF files, provide either a name or description for these elements.*

If you have several nested movie clips that serve a single purpose or convey one idea, ensure that you do the following:

- Group these elements in your SWF file.
- Provide a description for the parent movie clip.
- Make all the child movie clips inaccessible.

This is extremely important, or the screen reader tries to describe all the irrelevant nested movie clips, which will confuse the user, and might cause the user to leave your website. Make this decision whenever you have more than one object, such as many movie clips, in a SWF file. If the overall message is best conveyed using a single description, provide a description on one of the objects, and make all the other objects inaccessible to the screen reader.

Looping SWF files and applications cause screen readers to constantly refresh because the screen reader detects new content on the page. Because the reader thinks the content is updated, it returns to the top of the web page and starts rereading the content. Make inaccessible to screen readers any looping or refreshing objects that do not have to be reread.

***Note:** Do not type a description in the Description field of the Accessibility panel for instances (such as text) that the screen reader reads aloud.*

See also

"Using Flash to enter accessibility information for screen readers" on page 355

Using color

You must make decisions about using colors in an accessible file. You must not rely only on color to communicate particular information or directives to users. A color-blind user cannot operate a page if it asks to click on the blue area to launch a new page or the red area to hear music. Offer text equivalents on the page or in an alternate version to make your site accessible. Also, check that significant contrast exists between foreground and background colors to enhance readability. If you place light gray text on a white background, users cannot easily read it. Similarly, small text is difficult for many visitors to read. Using high-contrast and large or resizable text benefits most users, even those without impairments.

Ordering, tabbing, and the keyboard

Reading order and tabbing are important considerations for making accessible Flash websites. When you design an interface, the order that it appears on the page might not match the order in which the screen reader describes each instance. You can control and test reading order, as well as control tabbing in the SWF file.

Controlling reading order

The default reading order is not predictable and does not always match the placement of your assets or the visual layout of the page. Keeping the layout simple can help create a logical reading order without using ActionScript. However, you have more control over reading order if you use ActionScript and test the reading order in your SWF files.

***Important:** Do not miss ordering a single instance in your SWF file, or the reading order reverts to the default (and unpredictable) reading order.*

Controlling tabbing and content

Visitors who rely on screen readers to describe a site's content typically use tabbing and keyboard controls to navigate the operating system and web pages, because using the mouse is not useful when the screen cannot be seen. Use the `tabIndex` and `tabEnabled` properties with the movie clip, button, text field, or component instances to offer intelligent tabbing control in accessible SWF files. In addition to tabbing, you can use any key-press actions to navigate through the SWF file, but you must communicate that information using the Accessibility panel. Use the `Key` class in ActionScript to add key-press scripts to the SWF file. Select the object for which you want to use the key-press script, and add the shortcut key in the Shortcut field on the Accessibility panel. Add keyboard shortcuts to essential and frequently used buttons in your SWF file.

***Note:** In ActionScript 3.0, `tabIndex` and `tabEnabled` are properties of the `InteractiveObject` class. In ActionScript 2.0, they do not require a class reference.*

***Note:** Avoid invisible buttons in accessible SWF files, because screen readers do not recognize these buttons. (Invisible buttons are buttons for which you define only a hit area, the clickable region, for the button.)*

Many SWF files have a rapid succession of information, and screen readers frequently cannot keep up with this pace. Provide controls for the SWF file, letting the user use buttons to navigate through the file at their own pace, and letting them pause the process if necessary.

See also

"Creating accessibility with ActionScript" on page 362

Handling audio, video, and animation

When you provide audio narrations or video that contains speech, provide captions for those users who cannot hear. You can use text fields in Flash, import video that contains captions, or even use an XML caption file. You can use video cue points to specify when a text field should update text information at runtime.

For information on using Hi-Caption SE and the Hi-Caption Viewer component, see www.adobe.com/go/flash_extensions. This third-party extension lets you create captions that you save in an XML file and load into the SWF file at runtime, among other advanced controls. Alternatively, you can use cue points and a text field to display caption information.

See also

"Accessibility for hearing-impaired users" on page 354

Accessibility and extending Flash

With the extensibility layer in Flash, developers can create extensions that enable advanced authoring. This lets third-party companies develop extensions that involve accessibility. You have several options for validating your SWF files or adding captions.

For example, a validation tool can examine your SWF file for missing descriptions. It checks to see if a description is added for a group of instances, or if text has a label for the instance, and tells you about any problems. The tool also examines the reading order in your SWF file, and finds all instances that must be specified. You can specify reading order by using a dialog box after the SWF file is analyzed.

For information on the currently available third-party extensions, see www.adobe.com/go/flash_extensions.

Test files and make changes

Test any SWF file that is intended for use with screen readers. Test your SWF files when each new version of Flash Player is released, including minor revisions, and test it with the following:

- Window Eyes and JAWS for Windows screen readers, which handle SWF files differently, so you could get different results for the user experience.
- In a browser without a screen reader, and navigate through your site without using the mouse.
- Turn off your monitor and use only the screen reader to navigate your website.
- If you use audio narration, test your site without speakers.
- With several target visitors.

***Note:** You do not have to test different browsers, because the technology used to expose SWF files to screen readers (MSAA) is supported only by Internet Explorer on Windows.*

When listening to your SWF file using a screen reader, check the following points:

- Is the reading order correct?
- Do you have descriptions for shortcuts in your SWF file?
- Do you have adequate and complete descriptions for the elements in the interface?

- Do you have adequate descriptions for navigating the site's structure?
- Is the SWF file content read when it is updated or refreshed?
- If you change the context of any elements on the Stage (such as a button that changes from Play to Pause), is that change announced by the screen reader?

No official tool is available for validating SWF files, unlike HTML validation. However, some third-party tools exist to help you validate the file. For more information on these extensions, see www.adobe.com/go/flash_extensions.

Advertising with Flash

Using recommended dimensions

Use the Interactive Advertising Bureau (IAB) guidelines to set dimensions for your Flash advertisements. The following table lists the recommended Interactive Marketing Unit (IMU) ad formats measurements:

Type of advertisement	Dimensions (pixels)
Wide skyscraper	160 x 600
Skyscraper	120 x 600
Half-page ad	300 x 600
Full banner	468 x 60
Half banner	234 x 60
Micro bar	88 x 31
Button 1	120 x 90
Button 2	120 x 60
Vertical banner	120 x 240
Square button	125 x 125
Leaderboard	728 x 90
Medium rectangle	300 x 250
Square pop-up	250 x 250
Vertical rectangle	240 x 400
Large rectangle	336 x 280
Rectangle	180 x 150

When you create a FLA file from a template (Select File > New, and click the Templates tab), you see many of these sizes.

Creating SWF file advertisements

Use these guidelines when you create advertisements:

- Optimize your graphics. Make SWF file banner advertisements 15K or smaller.

- Create a GIF banner advertisement in Flash that is 12K or smaller.
- Limit looping banner advertisements to three repetitions. Many websites adopt the standardized file size recommendations as advertising specifications.
- Use the `GET` command to pass data between an advertisement and a server, and do not use the `POST` command. For more information on GET and POST, see the getURL function in *ActionScript 2.0 Language Reference*.

***Note:** Provide control to the user. If you add sound to an advertisement, also add a mute button. If you create a transparent Flash ad that hovers over a web page, provide a button to close the advertisement for its duration.*

See also

"Optimizing graphics and animation" on page 486

Tracking advertisements

Several leading advertising networks now support standardized tracking methods in Flash SWF files. The following guidelines describe the supported tracking methodology:

Create a button or movie clip button Use standardized dimensions outlined by the IAB. For a list of standardized dimensions, see the IAB website. For more information on creating a button in Flash, see "Create a button" on page 220.

Add a script to the button Executes when a user clicks the banner. You might use the `getURL()` function to open a new browser window. The following code snippets are two examples of ActionScript 2.0 code you might add to Frame 1 of the Timeline:

```
myButton_btn.onRelease = function(){
    getURL(clickTAG, "_blank");
};
```

You might add the following code to Frame 1 of the Timeline:

```
myButton_btn.onRelease = function() {
    if (clickTAG.substr(0, 5) == "http:") {
        getURL(clickTAG);
    }
};
```

The `getURL()` function adds the variable passed in the `object` and `embed` tags, and then sends the browser that is launched to the specified location. The server hosting the ad can track clicks on the advertisement. For more information on using the `getURL()` function, see *ActionScript 2.0 Language Reference*.

Assign clickTAG code for tracking Tracks the advertisement and helps the network serving the ad to track where the ad appears and when it is clicked.

The process is the standard way of creating an advertising campaign for a typical Flash advertisement. If you assign the `getURL()` function to the banner, you can use the following process to add tracking to the banner. The following example lets you append a variable to a URL string to pass data, which lets you set dynamic variables for each banner, instead of creating a separate banner for each domain. You can use a single banner for the entire campaign, and any server that is hosting the ad can track the clicks on the banner.

In the `object` and `embed` tags in your HTML, you would add code similar to the following example (where www.helpexamples.com is the ad network, and adobe.com is the company with an advertisement):

```
<EMBED src="your_ad.swf?clickTAG= http://helpexamples.com/tracking?http://www.adobe.com">
```

Add the following code in your HTML:

```
<PARAM NAME=movie VALUE="your_ad.swf?clickTAG =http:
//helpexamples.com/tracking?http://www.adobe.com">
```

For more information on advanced tracking techniques, see the Rich Media Advertising Center at www.adobe.com/go/rich_media_ads.

To download the Rich Media Tracking Kit, which includes examples and documentation, see www.adobe.com/go/richmedia_tracking.

To learn more about and download the Flash Ad Kit, which helps you deliver integrated and sophisticated advertisements, see www.adobe.com/go/learn_fl_flash_ad_kit.

Testing your ads

Test your SWF file ad on the most common browsers, especially the browsers that your target audience uses. Some users might not have Flash Player installed or they might have JavaScript disabled. Plan for these circumstances by having a replacement (default) GIF image or other scenarios for these users. For more information on detecting Flash Player, see "Set publish options for the Flash SWF file format" on page 420. Give the user control of the SWF file. Let the user control any audio in the ad. If the advertisement is a borderless SWF file that hovers over a web page, let the user close the advertisement immediately and for the duration of the ad.

For the latest information on Flash Player version penetration for different regions, go to www.adobe.com/go/fp_version_penetration.

Optimizing FLA files for SWF output

Optimize Flash documents

As your document file size increases, so does its download time and playback speed. You can take several steps to prepare your document for optimal playback. As part of the publishing process, Flash automatically performs some optimization on documents. Before exporting a document, you can optimize it further by using various strategies to reduce the file size. You can also compress a SWF file as you publish it. As you make changes, test your document by running it on a variety of computers, operating systems, and Internet connections.

For a video tutorial about optimizing Flash content, see www.adobe.com/go/vid0140.

See also

"Creating and publishing Flash Video" on page 300

Optimize documents

- Use symbols, animated or otherwise, for every element that appears more than once.
- Use tweened animations whenever possible when creating animation sequences. Tweened animations use less file space than a series of keyframes.
- Use movie clips instead of graphic symbols for animation sequences.
- Limit the area of change in each keyframe; make the action take place in as small an area as possible.
- Avoid animating bitmap elements; use bitmap images as background or static elements.

- Use mp3, the smallest sound format, whenever possible.

Optimize elements and lines

- Group elements.
- Use layers to separate elements that change during the animation from elements that do not.
- Use Modify > Shape > Optimize to minimize the number of separate lines that are used to describe shapes.
- Limit the number of special line types, such as dashed, dotted, ragged, and so on. Solid lines require less memory. Lines created with the Pencil tool require less memory than brush strokes.

Optimize text and fonts

- Limit the number of fonts and font styles. Use embedded fonts sparingly because they increase file size.
- For Embed Fonts options, select only the characters needed instead of including the entire font.

Optimize colors

- Use the Color menu in the Symbol Property inspector to create many instances of a single symbol in different colors.
- Use the Color panel (Window > Color) to match the color palette of the document to a browser-specific palette.
- Use gradients sparingly. Filling an area with gradient color requires about 50 bytes more than filling it with solid color.
- Use alpha transparency sparingly because it can slow playback.

Speed up document display

To speed up the document display, use commands in the View menu to turn off rendering-quality features that require extra computing and slow down document display.

None of these commands have any effect on how Flash exports a document. To specify the display quality of Flash documents in a web browser, use the `object` and `embed` parameters. The Publish command can do this for you automatically.

❖ Select View > Preview Mode, and select from the following options:

Outlines Displays only the outlines of the shapes in your scene and causes all lines to appear as thin lines. This makes it easier to reshape your graphic elements and to display complex scenes quickly.

Fast Turns off anti-aliasing and displays all the colors and line styles of your drawing.

Antialias Turns on anti-aliasing for lines, shapes, and bitmaps and displays shapes and lines so that their edges appear smoother on the screen. Draws more slowly than the Fast option. Anti-aliasing works best on video cards that provide thousands (16-bit) or millions (24-bit) of colors. In 16- or 256-color mode, black lines are smoothed, but colors might look better in Fast mode.

Antialias Text Smooths the edges of any text. Works best with large font sizes and can be slow with large amounts of text. This is the most common mode in which to work.

Full Renders all content on the Stage fully. Might slow down display.

See also

“Publishing overview” on page 418

Optimizing graphics and animation

Before you create optimized and streamlined animations or graphics, outline and plan your project. Make a target for the file size and length of the animation, and test throughout the development process.

Follow these guidelines to optimize graphics and animation:

- Avoid using gradients, because they require many colors and calculations to be processed, which is more difficult for a computer processor to render.
- For the same reason, keep the amount of alpha or transparency you use in a SWF file to a minimum.

Animating objects that include transparency is processor-intensive and should be kept to a minimum. Animating transparent graphics over bitmaps is a particularly processor-intensive kind of animation, and must be kept to a minimum or avoided completely.

***Note:** The best bitmap format to import into Flash is PNG, which is the native file format of Macromedia Fireworks from Adobe. PNG files have RGB and alpha information for each pixel. If you import a Fireworks PNG file into Flash, you retain some ability to edit the graphic objects in the FLA file.*

- Optimize bitmaps without overcompressing them. A 72-dpi resolution is optimal for the web. Compressing a bitmap image reduces file size, but compressing it too much compromises the quality of the graphic. Check that the settings for JPEG quality in the Publish Settings dialog box do not overcompress the image. Representing an image as a vector graphic is preferable in most cases. Using vector images reduces file size, because the images are made from calculations instead of many pixels. Limit the number of colors in your image while still retaining quality.

***Note:** Avoid scaling bitmaps larger than their original dimensions, because it reduces the quality of the image and is processor intensive.*

- Set the `_visible` property to `false` instead of changing the `_alpha` level to 0 or 1 in a SWF file. Calculating the `_alpha` level for an instance on the Stage is processor intensive. If you disable the instance's visibility, it saves CPU cycles and memory, which can give your SWF files smoother animations. Instead of unloading and possibly reloading assets, set the `_visible` property to `false`, which is less processor-intensive.
- Reduce the number of lines and points you use in a SWF file. Use the Optimize Curves dialog box (Modify > Shape > Optimize) to reduce the number of vectors in a drawing. Select the Use Multiple Passes option for more optimization. Optimizing a graphic reduces file size, but compressing it too much compromises its quality. However, optimizing curves reduces your file size and improves SWF file performance. Third-party options are available for specialized optimization of curves and points that yield different results.

To get the best results, try different ways of producing animated content, and test each of the options.

A higher frame rate (measured in frames per second, or *fps*) produces smooth animation in a SWF file but it can be processor-intensive, particularly on older computers. Test your animations at different frame rates to find the lowest frame rate possible.

For a sample of scripted animation, see the Flash Samples web page at www.adobe.com/go/learn_fl_samples. Download and decompress the Samples zip file and navigate to the ActionScript/Animation folder to access the sample.

See also

"Animation frame rate and performance" on page 487

"Video conventions" on page 469

Animation frame rate and performance

When you add animation to an application, consider the frame rate that you set your FLA file to. Frame rate can affect the performance of your SWF file and the computer that plays it. Setting a frame rate too high can lead to processor problems, especially when you use many assets or use ActionScript to create animation.

However, you also need to consider the frame rate setting, because it affects how smoothly your animation plays. For example, an animation set to 12 frames per second (fps) in the Property inspector plays 12 frames each second. If the document's frame rate is set to 24 fps, the animation appears to animate more smoothly than if it ran at 12 fps. However, your animation at 24 fps also plays faster than it does at 12 fps, so the total duration (in seconds) is shorter. Therefore, to make a 5-second animation using a higher frame rate, you must add additional frames to fill those five seconds than at a lower frame rate (and thus, raises the total file size of your animation). A 5-second animation at 24 fps typically has a higher file size than a 5-second animation at 12 fps.

***Note:** When you use an onEnterFrame event handler to create scripted animations, the animation runs at the document's frame rate, similar to if you created a motion tween on a timeline. An alternative to the onEnterFrame event handler is setInterval (see ActionScript 2.0 Language Reference). Instead of depending on frame rate, you call functions at a specified interval. Like onEnterFrame, the more frequently you use setInterval to call a function, the more resource intensive the animation is on your processor.*

Use the lowest possible frame rate that makes your animation appear to play smoothly at runtime, which helps reduce the strain on the end-user's processor. High frame rates (more than 30 to 40 fps) put a lot of stress on processors, and do not change the appearance of the animation much or at all at runtime.

Select a frame rate for your animation as early as possible in the development process. When you test the SWF file, check the duration, and the SWF file size, of your animation. The frame rate greatly affects the speed of the animation.

Filters and SWF file performance

If you use too many filters in an application, you can use large amounts of memory and cause Flash Player performance to suffer. Because a movie clip with filters attached has two bitmaps that are both 32-bit, these bitmaps can cause your application to use a significant amount of memory if you use many bitmaps. The computer's operating system might generate an out-of-memory error. On a modern computer, out-of-memory errors should be rare, unless you are using filter effects extensively in an application (for example, you have thousands of bitmaps on the Stage).

However, if you do encounter an out-of-memory error, the following occurs:

- The filters array is ignored.
- The movie clip is drawn using the regular vector renderer.
- No bitmaps are cached for the movie clip.

After an out-of-memory error occurs, a movie clip never attempts to use a filters array or a bitmap cache. Another factor that affects player performance is the value that you use for the quality parameter for each filter that you apply. Higher values require more CPU and memory for the effect to render, whereas setting the quality parameter to a lower value requires fewer computer resources. Avoid using an excessive number of filters, and use a lower quality setting when possible.

***Important:** If a 100 pixel by 100 pixel object is zoomed in once, it uses four times the memory since the content's dimensions are now 200 pixels by 200 pixels. If you zoom another two times, the shape is drawn as an 800 pixel by 800 pixel object which uses 64 times the memory as the original 100 pixel by 100 pixel object. Whenever you use filters in a SWF file, disable the zoom menu options from the SWF file's context menu.*

You can encounter errors if you use invalid parameter types. Some filter parameters also have a particular valid range. If you set a value that's outside of the valid range, the value changes to a valid value that's within the range. For example, quality should be a value from 1 to 3 for a standard operation, and can only be set to 0 to 15. Anything higher than 15 is set to 15.

Some constructors have restrictions on the length of arrays required as input parameters. If a convolution filter or color matrix filter is created with an invalid array (not the right size), the constructor fails and the filter is not created successfully. If the filter object is then used as an entry on a movie clip's filters array, it is ignored.

When using a blur filter, using values for blurX and blurY that are powers of 2 (such as 2, 4, 8, 16, and 32) can be computed faster and give a 20% to 30% performance improvement.

Bitmap caching and SWF file performance

Bitmap caching helps you enhance the performance of nonchanging movie clips in your applications. When you set the `MovieClip.cacheAsBitmap` or `Button.cacheAsBitmap` property to `true`, Flash Player caches an internal bitmap representation of the movie clip or button instance. This can improve performance for movie clips that contain complex vector content. All of the vector data for a movie clip that has a cached bitmap is drawn to the bitmap, instead of to the main Stage.

***Note:** The bitmap is copied to the main Stage as unstretched, unrotated pixels snapped to the nearest pixel boundaries. Pixels are mapped one-to-one with the parent object. If the bounds of the bitmap change, the bitmap is re-created instead of being stretched.*

For detailed information on caching button or movie clip instances see the following topics:

- About caching and scrolling movie clips with ActionScript in *Learning ActionScript 2.0 in Adobe Flash*
- Caching a movie clip in *Learning ActionScript 2.0 in Adobe Flash*

Use the `cacheAsBitmap` property with movie clips with mostly static content and that do not scale and rotate frequently. With such movie clips, using the `cacheAsBitmap` property can lead to performance improvements when the movie clip is translated (when its `x` and `y` position is changed).

Enabling caching for a movie clip creates a *surface*, which has several advantages, such as helping complex vector animations to render fast. In some situations, enabling caching does not improve performance, or even decrease it.

Overall performance of cached data depends on how complex the vector data of your instances are, how much of the data you change, and whether or not you set the opaqueBackground property. If you are changing small regions, the difference between using a surface and using vector data might be negligible. Test both scenarios with your work before you deploy the application.

When to use bitmap caching

The following are typical scenarios in which you might see significant benefits when you enable bitmap caching by optimizing vector graphics.

Complex background image An application that contains a detailed and complex background image of vector data. To improve performance, select the content, store it in a movie clip, and set the opaqueBackground property to true. The background is rendered as a bitmap and can be redrawn quickly, so that your animation plays faster.

Scrolling text field An application that displays a large amount of text in a scrolling text field. Place the text field in a movie clip that you set as scrollable with scrolling bounds (the scrollRect property), enabling fast pixel scrolling for the specified instance. When a user scrolls the movie clip instance, the scrolled pixels shift up and generate the newly exposed region instead of regenerating the entire text field.

Windowing system An application with a complex system of overlapping windows. Each window can be open or closed (for example, web browser windows). If you mark each window as a surface (set the cacheAsBitmap property to true), each window is isolated and cached. Users can drag the windows so that they overlap each other, and each window doesn't need to regenerate the vector content.

When to avoid using bitmap caching

Misusing bitmap caching can negatively affect your SWF file. When you develop a FLA file that uses surfaces, remember the following guidelines:

- Do not overuse surfaces (movie clips with caching enabled). Each surface uses more memory than a regular movie clip; only enable surfaces to improve rendering performance.
- A cached bitmap can use significantly more memory than a regular movie clip instance. For example, if the movie clip on the Stage is 250 pixels by 250 pixels, when cached it might use 250 KB instead of 1 KB when it's a regular (uncached) movie clip instance.
- Avoid zooming in on cached surfaces. If you overuse bitmap caching, a large amount of memory is consumed (see previous bullet), especially if you zoom in on the content.
- Use surfaces for movie clip instances that are largely static (nonanimating). You can drag or move the instance, but the contents of the instance should not animate or change a lot. For example, if you rotate or transform an instance, the instance changes between the surface and vector data, which is difficult to process and negatively affects your SWF file.
- If you mix surfaces with vector data, it increases the amount of processing that Flash Player (and sometimes the computer) needs to do. Group surfaces together; for example, when you create windowing applications.

Working with components in Flash Player

The component framework lets you add functionality to components, but it can potentially add considerable file size to an application. Components inherit from each other. One component adds size to your Flash document, but subsequent components that use the same framework do not necessarily add more size. As you add components to the Stage, the file size increases, but at some point, it levels off because components share classes and do not load new copies of those classes.

If you use multiple components that do not share the same framework, they might add substantial file size to the SWF file. For example, the XMLConnector component adds 17K to the SWF file, and TextInput components add 24K to your document. If you add the ComboBox component, it adds 28K, because it is not part of the framework of either previous component. Because the XMLConnector component uses data binding, the classes add 6K to the SWF file. A document that uses all these components has 77K before you add anything else to the file. Carefully consider your SWF file size when you add a new component to the document.

Components must exist in the parent SWF file's library. For example, an application must have a copy of the components it uses in its library, even if those components are required only by child SWF files that are loaded at runtime. This is necessary to ensure that the components function properly, and slightly increases the download time of the parent SWF file. However, the parent library isn't inherited or shared in the SWF files that you load into the parent. Each child SWF file must download to the application with its own copy of the same components.

When you are planning to publish a SWF file with backward compatibility, you must have a good understanding of which components have that capability. The following table provides information about component availability in different versions of Flash Player:

Components	Flash Player 6 (6.0.65.0) and earlier	Flash Player 6 (6.0.65.0)	Flash Player 7 and 8	Flash Player 9
ActionScript 3.0	Not supported	Not supported	Not supported	Supported
ActionScript 2.0	Supported	Supported	Supported	Supported
V2 UI component set	Not supported	Supported	Supported	Supported
Media components	Not supported	Not supported	Supported	Supported
Data components	Not supported	Not supported	Supported	Supported

Deselect the Optimize for Flash Player 6r65 option in Publish Settings for the V2 UI components to work.

Optimizing component styles and performance

When using ActionScript 2.0, one of the most processor-intensive calls in a component framework is the `setStyle` call. The `setStyle` call executes efficiently, but the call is intensive because of the way it is implemented. The `setStyle` call is not always necessary in all applications, but if you use it, consider its performance effect.

To enhance performance, you can change styles before they are loaded, calculated, and applied to the objects in your SWF file. If you can change styles before the styles are loaded and calculated, you do not have to call `setStyle`.

To improve performance when using styles, set properties on each object as objects are instantiated. When you dynamically attach instances to the Stage, set properties in `initObj` in the call that you make to `createClassObject()`, as the following ActionScript shows:

```
createClassObject(ComponentClass, "myInstance", 0, {styleName:"myStyle", color:0x99CCFF});
```

For instances that you place directly on the Stage, you can use `onClipEvent()` for each instance, or you can use subclasses (recommended). For information on subclasses, see About writing a subclass in *Learning ActionScript 2.0 in Adobe Flash*.

If you must restyle your components, you can improve efficiency in your application by using the Loader component. To implement several styles in different components, place each component in its own SWF file. If you change styles on the Loader component and reload the SWF file, the components in the SWF file are recreated. When the component is recreated, the cache of styles is emptied, and the style for the component is reset and referenced again.

***Note:** To apply a single style to all instances of a component in your SWF file, change the style globally using* `_global.styles.ComponentName`.

Using runtime shared libraries

You can sometimes improve download time by using runtime shared libraries. These libraries are usually necessary for larger applications or when numerous applications on a site use the same components or symbols. By externalizing the common assets of your SWF files, you do not download classes repeatedly. The first SWF file that uses a shared library has a longer download time, because both the SWF file and the library load. The library caches on the user's computer, and then all the subsequent SWF files use the library. This process can greatly improve download time for some larger applications.

Displaying special characters

Computer systems have a specific code page that is regional. For example, a computer in Japan has a different code page than a computer in England. Flash Player 5 and earlier versions relied on the code page to display text; Flash Player 6 and later versions use Unicode to display text. Unicode is more reliable and standardized for displaying text because it is a universal character set that contains characters for all languages. Most current applications use Unicode.

You can use Unicode escape sequences to display special characters in Flash Player 6. However, not all your characters display correctly if you do not load text that is UTF-8 or UTF-16 encoded (Unicode) or if you do not use a Unicode escape sequence to display the special character. For a set of Unicode code charts, see the Unicode web site at Unicode.org. For a list of commonly used escape sequences, see the table that follows in this section.

A non-Unicode application uses the operating system's code page to render characters on a page. In this case, the code page specifies the characters you see, so the characters appear correctly only when the code page on the user's operating system matches the application's code page. The code page that was used to create the SWF file needs to match the code page on the end user's computer. Using code pages is not a good idea for applications that an international audience might use; in this case, use Unicode instead.

Using `System.useCodepage` in your code forces the SWF file to use the system's code page instead of Unicode.

Only use this process when you are loading non-Unicode encoded text from an external location and when this text is encoded with the same code page as the user's computer. If both these conditions are true, the text appears without a problem. If both of these conditions are not true, use Unicode and a Unicode escape sequence to format your text. To use an escape sequence, add the following ActionScript 2.0 on Frame 1 of the Timeline:

```
this.createTextField("myText_txt", 99, 10, 10, 200, 25);
myText_txt.text = "this is my text, \u00A9 2004";
```

This ActionScript creates a text field, and enters text that includes a copyright symbol (©).

You can make a SWF file use the operating system's code page, which is controlled by the `useCodepage` property. When Flash exports a SWF file, it defaults to exporting Unicode text and `System.useCodepage` is set to `false`. You might encounter problems displaying special text, or text on international systems, where using the system's code page can seem to solve the problem of text incorrectly displaying. However, using `System.useCodePage` is always a last resort.

To use the system's code page, place the following line of AS 2.0 code on Frame 1 of the Timeline:

```
System.useCodepage = true;
```

Important: *A special character can appear only if the user's computer has the character included in the font that is being used. If you are not sure, embed the character or font in the SWF file.*

The following table contains a number of commonly used Unicode escape sequences.

Character description	**Unicode escape sequence**
em-dash (—)	\u2014
registered sign (®)	\u00AE
copyright sign (©)	\u00A9
trademark sign (™)	\u2122
Euro sign (€)	\u20AC
backslash (\)	\u005C

Character description	Unicode escape sequence
forward slash (/)	\u002F
open curly brace ({)	\u007B
close curly brace (})	\u007D
greater than (<)	\u003C
less than (>)	\u003E
asterisk (*)	\u002A

Test document download performance

Flash Player attempts to meet the frame rate you set; the actual frame rate during playback can vary on different computers. If a document that is downloading reaches a particular frame before the frame's required data is downloaded, the document pauses until the data arrives.

To view downloading performance graphically, use the Bandwidth Profiler, which shows how much data is sent for each frame according to the modem speed you specify.

In simulating the downloading speed, Flash uses estimates of typical Internet performance, not the exact modem speed. For example, if you select to simulate a modem speed of 28.8 Kbps, Flash sets the actual rate to 2.3 Kbps to reflect typical Internet performance. The profiler also compensates for the added compression support for SWF files, which reduces the file size and improves streaming performance.

When external SWF files, GIF and XML files, and variables are streamed into a player by using ActionScript calls such as `loadMovie` and `getUrl`, the data flows at the rate set for streaming. The stream rate for the main SWF file is reduced based on the reduction of bandwidth that the additional data requests cause. Test your document at each speed and on each computer that you plan to support to ensure that the document doesn't overburden the slowest connection and computer for which it is designed.

You can also generate a report of frames that are slowing playback and then optimize or eliminate some of the content in those frames.

To change the settings for the SWF file created using the Test Movie and Test Scene commands, use File > Publish Settings.

See also

"Optimize Flash documents" on page 484

"Publishing overview" on page 418

"Debugging ActionScript 1.0 and 2.0" on page 400

"Debugging ActionScript 3.0" on page 410

Test download performance

1 Do one of the following:

- Select Control > Test Scene or Control > Test Movie.

If you test a scene or document, Flash publishes the current selection as a SWF file using the settings in the Publish Settings dialog box. The SWF file opens in a new window and begins playing immediately.

- Select File > Open, and select a SWF file.

2 Select View > Download Settings, and select a download speed to determine the streaming rate that Flash simulates. To enter a custom user setting, select Customize.

3 When viewing the SWF file, select View > Bandwidth Profiler to show a graph of the downloading performance.

The left side of the profiler displays information about the document, its settings, its state, and streams, if any are included in the document.

The right section of the profiler shows the Timeline header and graph. In the graph, each bar represents an individual frame of the document. The size of the bar corresponds to that frame's size in bytes. The red line beneath the Timeline header indicates whether a given frame streams in real time with the current modem speed set in the Control menu. If a bar extends above the red line, the document must wait for that frame to load.

4 Select View > Simulate Download to turn streaming off or on.

If you turn streaming off, the document starts over without simulating a web connection.

5 Click a bar on the graph to show settings for the corresponding frame in the left window and stop the document.

6 If necessary, adjust the view of the graph by taking one of the following actions:

- Select View > Streaming Graph to show which frames cause pauses.

This default view displays alternating light and dark gray blocks that represent each frame. The side of each block indicates its relative byte size. The first frame stores a symbol's contents, so it is often larger than other frames.

- Select View > Frame by Frame Graph to display the size of each frame.

This view helps you see which frames contribute to streaming delays. If any frame block extends above the red line in the graph, Flash Player stops playback until the entire frame downloads.

7 Close the test window to return to the authoring environment.

After you set up a test environment using the Bandwidth Profiler, you can open any SWF file directly in the test environment. The file opens in a Flash Player window, using the Bandwidth Profiler and other selected viewing options.

Generate a final report

1 Select File > Publish Settings, and click the Flash tab.

2 Select Generate Size Report.

3 Click Publish.

Flash generates a text file with the .txt extension. (If the document file is myMovie.fla, the text file is myMovie Report.txt.) The report lists the size of each frame, shape, text, sound, video and ActionScript script by frame.

Tips for creating content for mobile devices

Creating Flash content for use on mobile devices

To create Flash content for mobile devices, follow some basic principles. For example, Flash developers often avoid extremely complex artwork and excessive tweening or transparency.

Flash Lite developers face additional challenges because performance on mobile devices varies greatly. If content must be published to many different devices, developers sometimes have to author for the lowest common denominator.

Optimizing mobile content requires making trade-offs. For example, one technique may make the content look better, while another results in better performance. As you evaluate these trade-offs, you will be going back and forth repeatedly between testing in the emulator and testing on the target device. You must see your content on the actual device to evaluate the trueness of colors, text readability, physical interactions, UI responsiveness, and other aspects of the real mobile experience.

Flash Lite guidelines for animation in mobile devices

When creating animated content for a mobile devices, keep device CPU limitations in mind. Following these guidelines can help prevent Flash Lite content from running slowly:

- When creating a new Flash Lite file, check that the document is set up correctly. Although Flash files scale smoothly, performance can suffer if the file is not running at its native Stage size and has to scale in the player. Try to set the document Stage size to match the resolution of the target device. Also set the Flash Player to the correct version of Flash Lite and select an appropriate device profile in Device Central.
- Flash Lite can render vector graphics at low, medium, and high quality. The higher the rendering quality, the more smoothly and accurately Flash Lite renders vector graphics and the greater the demand on the device's CPU. To provide complex animation, experiment with changing the player's quality setting and then thoroughly test the SWF file. To control the rendering quality of a SWF file, use the `_quality` property or the `SetQuality` command. Valid values for the `_quality` property are `LOW`, `MEDIUM`, and `HIGH`.
- Limit the number of simultaneous tweens. Reduce the number of tweens, or sequence the animation so that one begins when another ends.
- Use transparency (alpha) effects on symbols sparingly because they are CPU intensive. In particular, avoid tweening symbols with alpha levels that are not fully opaque (less than 100%).
- Avoid CPU-intensive visual effects, such as large masks, extensive motion, alpha blending, extensive gradients, and complex vectors.
- Experiment with combinations of tweens, keyframe animations, and ActionScript-driven movement to produce the most efficient results.
- Rendering vector ovals and circles is much more memory intensive than rendering quadrangles. Using round and oval strokes also greatly increases CPU use.
- Test animations frequently on actual target devices.
- When Flash draws an animated region, it defines a rectangular bounding box around the area. Optimize the drawing by making that rectangle as small as possible. Avoid overlapping tweens, because Flash interprets the merged area as a single rectangle, resulting in a larger total region. Use Flash's Show Redraw Region feature to optimize the animation.
- Avoid using `_alpha = 0` and `_visible = false` to hide on-screen movie clips. If you simply turn a movie clip's visibility off or change its alpha to zero, it is still included in line-rendering calculations, which can affect performance.
- Similarly, do not try to hide a movie clip by obscuring it behind another piece of artwork. It will still be included in the player's calculations. Instead, move movie clips completely off the Stage or remove them by calling `removeMovieClip`.

Flash Lite bitmap and vector graphics in mobile devices

Flash Lite can render both vector and bitmap graphics. Each type of graphic has its advantages and disadvantages. The decision to use vector rather than bitmap graphics is not always clear and often depends on several factors.

Vector graphics are compactly represented in SWF files as mathematical equations and rendered at run time by the Flash Lite player. In contrast, bitmap graphics are represented as arrays of picture elements (pixels), which require more bytes of data. Therefore, using vector graphics in a file can help reduce file size and memory usage.

Vector graphics also maintain their smooth shapes when scaled in size. Bitmap images can appear boxy, or pixelated, when scaled.

Compared to bitmaps, vector graphics require more processing power to render, especially vector graphics that have many complex shapes and fills. Consequently, heavy use of vector shapes can sometimes reduce overall file performance. Because bitmap graphics do not require as much processing time to render as vector graphics, they are better choice for some files, for example, a complex road map meant to be animated and scrolled on a mobile phone.

Keep these considerations in mind:

- Avoid using outlines on vector shapes. Outlines have an inner and outer edge (fills have only one) and are twice the work to render.
- Corners are simpler to render than curves. When possible, use flat edges, especially with very small vector shapes.
- Optimization is especially helpful with small vector shapes such as icons. Complex icons may lose their details upon rendering, and the work of rendering the details is wasted.
- As a general rule, use bitmaps for small, complex images (such as icons) and vector graphics for larger and simpler ones.
- Import bitmap graphics at the correct size; don’t import large graphics and scale them down in Flash, because this wastes file size and run-time memory.
- The Flash Lite player does not support bitmap smoothing. If a bitmap is scaled or rotated, it will have a chunky appearance. If it is necessary to scale or rotate a graphic, consider using a vector graphic instead.
- Text is essentially just a very complex vector shape. Of course, text is often critical, so it can rarely be avoided entirely. When text is needed, avoid animating it or placing it over an animation. Consider using text as a bitmap. For multiline dynamic and input text, the line break of the text string is not cached. Flash breaks lines at run time and recalculates the breaks every time the text field needs to be redrawn. Static text fields are not problematic, because the line breaking is precalculated at compile time. For dynamic content, using dynamic text fields is unavoidable, but when possible, consider using static text fields instead.
- Minimize the use of transparency in PNG files; Flash must calculate redraws even for the transparent portions of the bitmap. For example, with a transparent PNG file that represents a foreground element, don't export the transparent PNG at the full size of the screen. Instead, export it at the actual size of the foreground element.
- Try to group bitmap layers together and vector layers together. Flash needs to implement different renderers for bitmap and vector content, and switching between renderers takes time.

Set compression of Flash Lite bitmaps for mobile devices

When using bitmaps, you can set image-compression options (on a per-image basis or globally for all bitmap images) that reduce SWF file size.

Set compression options for an individual bitmap file

1 Start Flash and create a document.

2 Select a bitmap in the Library window.

3 Right-click (Windows) or Control-click (Macintosh) the bitmap icon in the Library window, and select Properties from the context menu to open the Bitmap Properties dialog box.

4 In the Compression pop-up menu, select one of the following options:

- Select the Photo (JPEG) option for images with complex color or tonal variations, such as photographs or images with gradient fills. This option produces a JPEG file. Select the Use Imported JPEG Data check box to use the default compression quality specified for the imported image. To specify a new quality compression setting, deselect Use Imported JPEG Data and enter a value between 1 and 100 in the Quality text box. A higher setting produces an image of higher image, but also a larger file, so adjust the value accordingly.
- Select the Lossless (PNG/GIF) option for images with simple shapes and a few colors. This option compresses the image using lossless compression, which discards no data.

5 Click Test to determine the results of the file compression.

Compare the original file size to the compressed file size to decide whether the selected compression setting is acceptable.

Set compression for all bitmap images

1 Select File > Publish Settings, and then click the Flash tab to display compression options.

2 Adjust the JPEG quality slider, or enter a value. A higher JPEG quality value produces an image of higher image quality but a larger SWF file. A lower image quality produces a smaller SWF file. Try different settings to determine the best trade-off between size and quality.

Optimizing Flash Lite frames for mobile devices

- Most devices that support Flash Lite play back content at about 15 to 20 frames per second (fps). The frame rate can be as low as 6 fps. During development, set the document frame rate to approximate the playback speed of the target device. This shows how the content will run on a device with limited performance. Before publishing a final SWF file, set the document frame rate to at least 20 fps or higher to avoid limiting performance in case the device supports a higher frame rate.
- When using `gotoAndPlay`, remember that every frame between the current frame and the requested frame needs to be initialized before Flash plays the requested frame. If many of these frames contain different content, it could be more efficient to use different movie clips rather than using the Timeline.
- Although preloading all content by putting it at the beginning of the file makes sense on the desktop, preloading on a mobile device can delay file startup. Space content throughout the file so that movie clips are initialized as they are used.

Optimizing ActionScript for Flash Lite content on mobile devices

Because of the processing speed and memory limitations on most mobile devices, follow these guidelines when developing ActionScript for Flash Lite content used on mobile devices:

- Keep the file and its code as simple as possible. Remove unused movie clips, delete unnecessary frame and code loops, and avoid too many frames or extraneous frames.
- Using `FOR` loops can be expensive because of the overhead incurred while the condition is checked with each iteration. When the costs of the iteration and the loop overhead are comparable, execute multiple operations individually instead of using a loop. The code may be longer, but performance will improve.
- Stop frame-based looping as soon as it is no longer needed.
- When possible, avoid string and array processing because it can be CPU-intensive.

- Always try to access properties directly rather than using ActionScript getter and setter methods, which have more overhead than other method calls.
- Manage events wisely. Keep event listener arrays compact by using conditions to check whether a listener exists (is not `null`) before calling it. Clear any active intervals by calling `clearInterval`, and remove any active listeners by calling `removeListener` before removing content using `unloadapplication` or `removeapplicationClip`. Flash does not re-collect SWF data memory (for example, from intervals and listeners) if any ActionScript functions are still referring to the SWF data when a movie clip is unloaded.
- When variables are no longer needed, delete them or set them to `null`, which marks them for garbage collection. Deleting variables helps optimize memory use during run time, because unneeded assets are removed from the SWF file. It is better to delete variables than to set them to `null`.
- Explicitly remove listeners from objects by calling `removeListener` before garbage collection.
- If a function is being called dynamically and passing a fixed set of parameters, use `call` instead of `apply`.
- Make namespaces (such as paths) more compact to reduce startup time. Every level in the package is compiled to an `IF` statement and causes a new `Object` call, so having fewer levels in the path saves time. For example, a path with the levels `com.xxx.yyy.aaa.bbb.ccc.funtionName` causes an object to be instantiated for `com.xxx.yyy.aaa.bbb.ccc`. Some Flash developers use preprocessor software to reduce the path to a unique identifier, such as `58923409876.functionName`, before compiling the SWF code.
- If a file consists of multiple SWF files that use the same ActionScript classes, exclude those classes from select SWF files during compilation. This can help reduce file download time and run-time memory requirements.
- Avoid using `Object.watch` and `Object.unwatch`, because every change to an object property requires the player to determine whether a change notification must be sent.
- If ActionScript code that executes on a keyframe in the timeline requires more than 1 second to complete, consider splitting up that code to execute over multiple keyframes.
- Remove `trace` statements from the code when publishing the SWF file. To do this, select the Omit Trace Actions check box on the Flash tab in the Publish Settings dialog box.
- Inheritance increases the number of method calls and uses more memory: a class that includes all the functionality it needs is more efficient at run time than a class that inherits some of its functionality from a superclass. Therefore, you may need to make a design trade-off between extensibility of classes and efficiency of code.
- When one SWF file loads another SWF file that contains a custom ActionScript class (for example, `foo.bar.CustomClass`) and then unloads the SWF file, the class definition remains in memory. To save memory, explicitly delete any custom classes in unloaded SWF files. Use the `delete` statement and specify the fully qualified class name, such as: `delete foo.bar.CustomClass.`
- Limit the use of global variables, because they are not marked for garbage collection if the movie clip that defined them is removed.
- Avoid using the standard user interface components (available in the Components panel in Flash). These components are designed to run on desktop computers and are not optimized to run on mobile devices.
- Whenever possible, avoid deeply nested functions.
- Avoid referencing nonexistent variables, objects, or functions. Compared to the desktop version of Flash Player, Flash Lite 2 looks up references to nonexistent variables slowly, which can significantly affect performance.
- Avoid defining functions using anonymous syntax. For example, `myObj.eventName = function{ ...}`. Explicitly defined functions are more efficient, such as `function myFunc { ...}; my Obj.eventName = myFunc;`.

- Minimize the use of Math functions and floating-point numbers. Calculating these values slows performance. If you must use the Math routines, consider precalculating the values and storing them in an array of variables. Retrieving the values from a data table is much faster than having Flash calculate them at run time.

Managing Flash Lite file memory for mobile devices

Flash Lite regularly clears from memory any objects and variables that a file no longer references. This is known as garbage collection. Flash Lite runs its garbage-collection process once every 60 seconds, or whenever usage of file memory increases suddenly by 20% or more.

Although you cannot control how and when Flash Lite performs garbage collection, you can still free unneeded memory deliberately. For timeline or global variables, use the `delete` statement to free the memory that ActionScript objects use. For local variables—for example, a variable defined within a function definition—you can't use the `delete` statement to free an object's memory, but you can set to `null` the variable that references the object. This frees the memory that the object uses, provided there are no other references to that object.

The following two code examples show how to free memory that objects use by deleting the variable that references those objects. The examples are identical, except that the first example creates a timeline variable and the second creates a global variable.

```
// First case: variable attached to a movie or
// movie clip timeline
//
// Create the Date object.
var mcDateObject = new Date();
// Returns the current date as a string.
trace(mcDateObject);
// Delete the object.
delete mcDateObject;
// Returns undefined.
trace(mcDateObject);
//
// Second case: global variable attached to a movie or
// movie clip timeline
//
// Create the Date object.
_global.gDateObject = new Date();
// Returns the current date as a string.
trace(_global.gDateObject);
// Delete the object.
delete _global.gDateObject;
// Returns undefined.
trace(_global.gDateObject);
```

As mentioned previously, you can't use the `delete` statement to free memory that a local function variable uses. Instead, set the variable reference to `null`, which has the same effect as using `delete`.

```
function func()
{
    // Create the Date object.
    var funcDateObject = new Date();
    // Returns the current date as a string.
    trace(funcDateObject);
    // Delete has no effect.
    delete funcDateObject;
    // Still returns the current date.
    trace(funcDateObject);
    // Set the object reference to null.
    funcDateObject = null;
    // Returns null.
    trace(funcDateObject);
}
// Call func() function.
func();
```

Loading data for mobile devices in Flash Lite

When developing files for mobile devices, minimize the amount of data you attempt to load at one time. If you are loading external data into a Flash Lite file (for example, using `XML.load`), the device's operating system may generate a "memory failure" error if insufficient memory is allocated for the incoming data. This situation can occur even if the total amount of remaining memory is sufficient.

For example, suppose a file attempts to load an XML file that's 100 KB, but the device's operating system has allocated only 30 KB to handle that incoming data stream. In this case, Flash Lite displays an error message to the user, indicating that not enough memory is available.

To load large amounts of data, group the data in smaller pieces—for example, in several XML files—and make several data-loading calls for each piece. The size of each piece of data, and therefore the number of data-loading calls you need to make, varies by device and file. To determine an appropriate balance between the number of data requests and the likelihood of a memory failure, test files on a variety of target devices.

For optimum performance, avoid loading and parsing XML files if possible. Instead, store data in simple name/value pairs and load the data from a text file using `loadVars` or from precompiled SWF files.

Exclude classes from compilation for Flash Lite

To reduce the size of a SWF file, consider excluding classes from compilation but retaining the ability to access and use them for type checking. For example, try this if you are developing a file that uses multiple SWF files or shared libraries, especially those that access many of the same classes. Excluding classes helps avoid duplicating classes in those files.

1 Create a new XML file.

2 Name the XML file FLA_filename_exclude.xml, where FLA_filename is the name of the FLA file without the .fla extension. For example, if the FLA file is sellStocks.fla, the XML filename must be sellStocks_exclude.xml.

3 Save the file in the same directory as the FLA file.

4 Place the following tags in the XML file:

```
excludeAssets>)
    <asset name=”className1” />)
    <asset name=”className2” />
</excludeAssets>
```

The values specified for the name attributes in the `<asset>` tags are the names of classes that should be excluded from the SWF file. Add as many as required for the file. For example, the following XML file excludes the `mx.core.UIObject` and `mx.screens.Slide` classes from the SWF file:

```
<excludeAssets>)
    <asset name=”mx.core.UIObject” />)
    <asset name=”mx.screens.Slide” />
</excludeAssets>
```

Index