

# How to optimize Flash content

As with all web content, it is important to keep Adobe Flash Professional file sizes as small as possible. Even though connection speeds have increased in the past few years, keeping file sizes small continues to be important, because document size can also affect content performance, and content is now delivered on a range of devices. This process is known as optimizing Flash content.

The following general principles can help you optimize Flash content:

- Use symbols, animated or otherwise, for every element that appears more than once.
- When creating animation sequences, use tweened animations whenever possible. These animations use less file space than a series of keyframes.
- For animation sequences, use movie clips instead of graphic symbols.
- 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.

## Optimizing images

Images present one of the best opportunities for optimizing file size in Flash.

Certain fills and effects can increase the size of your Flash movie. Use the following tips to optimize these movie features:

- Use the Color Effect section in the Symbol Properties panel to create many different-colored instances of a single symbol instead of creating multiple symbols.
- Use gradients sparingly. Minimize the number of simultaneous gradients in any given frame. Filling an area with gradient color requires about 50 bytes more than filling it with solid color. Gradients are also more complex for a computer processor to handle than a solid color.
- Use alpha transparency sparingly. It can slow playback.

## Optimizing bitmaps

*Bitmap* images take up more space than vector images (vector is Flash's normal format). Bitmap images are imported into Flash. JPEGs, GIFs, and BMPs are all bitmapped. The best bitmap format to import into Flash is PNG, which is the native file format of Adobe Fireworks. All photos—which are usually in JPEG form—are bitmapped. When you create an animation with bitmap images, Flash must store one bitmap image in each frame of the animation. In addition, bitmap images, unlike vector images, do not scale or rotate well on the screen; and bitmap images must be rendered each time they change positions on the screen. This often entails changing large areas of the movie and slowing down playback as these areas are continually redrawn.

If you feel your Flash movie must include bitmaps, use the following tips to optimize your movie:

- Use bitmap images sparingly.
- Avoid animating bitmap images.
- Convert the bitmap image into a symbol if it appears in more than one keyframe.

## Use the Trace Bitmap command

The Trace Bitmap command (Modify > Bitmap > Trace Bitmap) lets you convert a bitmap to a vector image. This works best with simple images made up of mostly large blocks of color. If the bitmap contains complex shapes and many colors, the converted vector graphic can have a larger file size than the original bitmap. Try a variety of settings in the Trace Bitmap dialog box to find a balance between file size and image quality.

## Compression options

Compression helps reduce the total file space occupied by bitmap images. JPEG compression is the default compression for all bitmap images published from Flash. In the Publish Settings dialog box, you can adjust the value in the JPEG Quality field from 1 (most compression, lowest quality) to 100 (least compression, highest quality). Because JPEG compression can result in a loss of image data, test images with different JPEG quality values to achieve the best balance between compression and image quality for your particular movie.

## Optimizing sound

Sound is another factor that can increase Flash file size. There are several ways to use sound efficiently in a movie and keep file size down.

- Use MP3 compression, the smallest sound format, whenever possible.
- Keep sounds short.
- Use the lowest bit-depth and sample rate acceptable to achieve the smallest data size. Try to find a happy medium between quality and data size.
- Loop short sounds to create background music.
- Looping smaller event sounds can achieve longer audio playback because a small amount of sound data is downloaded once and used repeatedly.
- Do not set streaming sound to loop.
- Use the Event synch option to deliver the sound data all at once as needed for playback. Use Stream synch to deliver the sound data over the specified series of frames. Try both approaches. Then watch the size report closely for clogging of data near keyframes, and use the preloading techniques to trim down the amount of data needed per frame or per second. You can generate a size report for a Flash movie when viewing the Flash (.swf) settings in the Publish Settings dialog box (File > Publish Settings, click Flash (.swf.), select Generate Size Report in the Advanced section).

For more information about best practices in Flash, see the “Best Practices” section in Using Flash (Help > Flash Help).

## Setting audio compression options

In the following steps, you set audio to *stream* and add the audio to the timeline.

To set audio compression options:

1. Import audio into the sound layer of your Flash document.
2. In the Properties panel, set Sync to Stream (**Figure 1**).

Although streaming means losing some audio quality, it creates a smaller SWF file and synchronizes the audio to the object movements on the Stage.

If your sound file is longer than the elapsed time in the Timeline panel (**Figure 2**), you'll need to add more frames in the sound layer to hear the entire sound file.

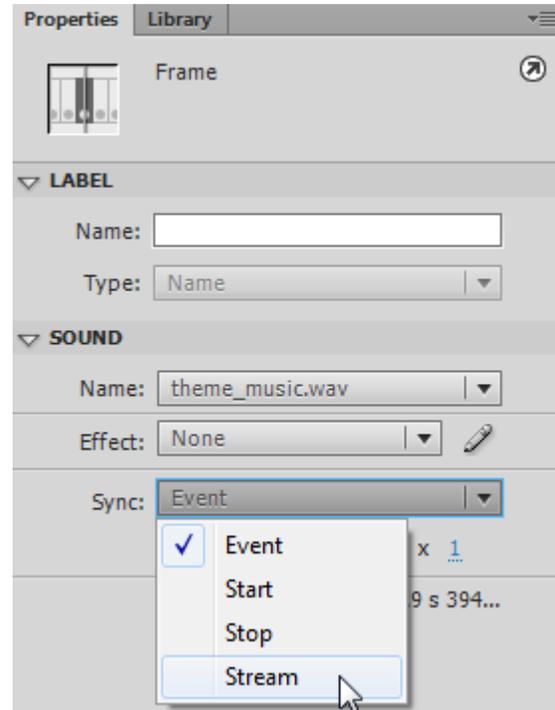
3. Use the frame rate shown at the bottom of the Timeline panel (**Figure 2**) and the length of the audio to calculate how many frames are required to play the entire sound file. For example, if your sound file is five seconds, and the frame rate is 24 fps, the timeline must extend to at least 120 frames.

You can add the new frames by inserting a single frame further down the timeline.

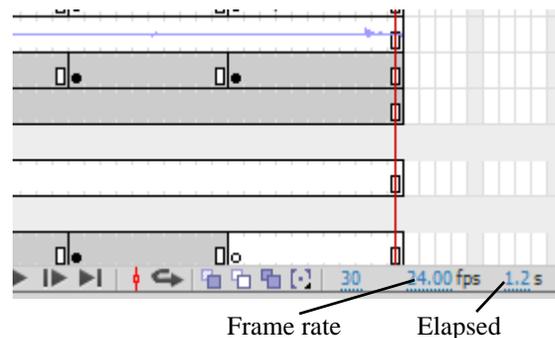
4. Based on the duration of the sound clip and the calculation you made in step 3, select the frame in the timeline where the sound file will end, and extend the timeline by selecting Insert Timeline > Frame.

All the audio will be visible in the timeline.

**Note:** Do not use Insert > Keyframe, because a new keyframe would stop the audio before it is finished.



**Figure 1** Sound section of the Properties panel



**Figure 2** Timeline panel