

# Performance Tips for Flash

This tutorial provides tips on how to speed up your Flash games.

## More Info

---

Sometimes slow-downs are caused by inefficient programming. But most of the time, they're caused by elaborate vector graphics, or an over-use of realtime effects like blur and glow.



## Convert vector graphics into pixel graphics

---

If the graphics won't be changing size, you can even make them bitmaps. That way, they barely require any processing, no matter how detailed they are.

This is because vector graphics are calculated and drawn in real-time every frame. Every line, fill, gradient, and layer needs to be figured out. Bitmap images are simply copied from memory.

Note: Always put your images into symbols. Never use images directly in an animation. They won't tween anyway.

## Reduce loops in code

---

When slow-down IS caused by code, it's usually because of loops. Such as when checking for collisions or searching for something.

The key is to figure out ways to reduce the size of "for" loops, or even look for some alternative ways that don't use looping at all. For instance, the collision-checking example below doesn't require any searching or hitTest() functions to find the walls. Instead it uses a 2-dimensional array and simply checks a location within it. This is useful for colliding with stationary blocks in a level.

If you're checking for collision with every character, figure out ways to exclude movieClips you're not looking for. Check only the things you need to check.

One way to do this is to make an array of references to each movieClip you wish to check, and simply loop through that array. The first step in collision checking is usually figuring out what to ignore.

[Download Source](#)

Or find ways to reduce the number of collision checks. For example, have characters check for blocks, instead of having blocks check for characters. Since there are fewer characters than blocks, it'll use fewer collision checks.

## Use image sequences and tweens instead of code

---

When possible, animate things with image sequences or tweens. It's easier for Flash to process built-in features, than it is to process Actionscript.

## Use a "finite state machine"

---

Separate the player's code into multiple small programs. And run only one at a time. Such as a "stand" program, a "walk" program, a "jump" program, etc...

This is called a "Finite State Machine"  
It's very efficient, and can also make things MUCH easier to program.

## Don't detect attacks

---

Have sprites "tell" each other that they're being attacked, via function calls, instead of each character trying to detect when it's being hit. Again, it's easier to program.