

MULTIMEDIA DEVELOPMENT AXIOMS

GLOBAL

- **Simplicity**
- **Clarity**
- **Generality**

DEFINITION AND SPECIFICATION

Use the right tool for the job.

Certain software tools have strengths in a particular area, but weaknesses in others. Flash is an excellent tool for creating animations and for implementing a Web-based interface. Its ability to shrink projects down to a downloadable size while maintaining high visual quality is what distinguishes it from other tools. But with all of this capability, Flash is not as strong as Director as an authoring tool overall. Director's position as a standard in the multimedia industry derives from its media integration and multimedia authoring capabilities. Most multimedia producers create media in other applications designed specifically for creating images, text, video, or sounds. Director integrates all those separate media types into a single presentation or project. Media integration includes collecting, managing, and synchronizing all the content of a project. Although Director shares many capabilities with other animation and presentation programs, it is its authoring capacity that distinguishes it from other applications.

The ambidextrous designer is a good designer.

There are at least three ways of doing anything in most multimedia tools. Most tasks in Flash, for example, can be accomplished by choosing an option from a menu, pressing a key combination on the keyboard, or pressing a graphical button attached to a window. It is advantageous to learn keyboard commands early on for several reasons. Using the keyboard saves time and encourages the designer to use both hands to navigate the Flash environment.

Save early and often.

Save your Flash file after every significant change but save it as different versions of the same movie, using the Save As command. Saving in this fashion allows restoration of earlier versions and protects against the loss of work. A common naming convention has the name of the Flash movie followed by a multi-digit number, indicating its version number. "KillerApp.01", "KillerApp.02", or "KillerApp.66", for example are successive versions of the same project. Storage space is "free" on a temporary basis, so keep all or most of the versions of all files until the very end of a project. Hard drives, after all, are designed to be filled. Recreating work because a file was deleted to make disk space is no fun. Most of us fail to learn this lesson until it's too late. At the conclusion of a project, the organization and the use of naming conventions permits a quick determination of what to save and what to discard.

DESIGNING THE PROJECT

Multimedia demands clarity of purpose.

It is critically important to have an idea of the whole multimedia project and the relationship of its parts rather than a “whiz-bang” idea for a single component. Without clarity of purpose or a message in mind, multimedia becomes so many “bells and whistles”. The project illustrates technique rather than communicates content. Technical pyrotechnics are interesting for only so long. The user ultimately responds with a “so what?” and moves on.

Multimedia require planning.

Having settled on a purpose or a message, multimedia designers engage in several planning steps to determine audience, delivery format, costs, tools, and content. Failure to plan in detail results in lost time, increased costs, missed deadlines, and general aggravation and stress. In fact, designers may spend more time in planning and brainstorming than in actual execution of pieces of art, music, video, or authoring.

Multimedia design and implementation call for organization.

A multimedia project may have dozens or even hundreds of separate media elements, so it is important to keep all of these pieces arranged by developing a file naming and disk storage system. Designers and their partners need to know what something is and where it is. This is especially important as time passes and familiarity with the project wanes. “What was ‘Untitled23.PCT’ anyway?” is not the type of question you want to have to ask in the middle of a project.

PROTOTYPING AND IMPLEMENTING THE DESIGN

Be patient.

Multimedia design and implementation are time-consuming. Despite vendors' claims that multimedia can be put together in a trice, multimedia programs generally have steep learning curves. Learning something new takes patience and concentration. Donald Norman, in *Things That Make Us Smart* reminds us that it takes three years to become expert in a field. Once a designer achieves fluency in the programs, multimedia still demands time.

Slow and steady wins the race.

Multimedia is detailed, precise work. As politicians and others are fond of saying, "The devil is in the details." The same can be said of multimedia. Attention to detail and precision are the stuff of multimedia. As later chapters explain, computers do exactly what they are told to do. Remember: the human eye can detect when a static multimedia element moves a pixel or two.

"It's déjà vu all over again."

Multimedia is iterative. That is to say, a designer tries an animation or set of actions, tests it, revises it or corrects errors, tests it again, and perhaps saves it. The process begins again: trying something else, testing it, revising or editing it, and so on. Simply put, multimedia work is repetitive.

ANIMATION

Less is more.

Simplify. Eliminate all unimportant detail in an animation and concentrate on what lines and movements are absolutely essential. Make those movements lean and the images or characters uncluttered. Use only enough animation to convey the message or mood.

Small is beautiful, especially in animation.

Animate small things rather than large things. Every time an object moves, the screen must be redrawn. The bigger the object, the greater the computing power necessary to redraw the screen the the greater the time involved. The greater the time, the choppier or jerkier the animation becomes.

Animate sequentially.

Generally, it is advantageous to animate objects in sequence, one after the other. This is especially true when the objects are medium-sized. The time between the movements will almost be imperceptible, and most computers can handle this task smoothly. You can also animate several small objects simultaneously. Small things move faster; large things must move more slowly.

WORKING WITH MEDIA ELEMENTS

Using symbols is a good strategy.

Flash's ability to work with symbols is advantageous in a number of ways. First, it reduces the number of individual media elements and assists a designer in staying within hardware and software constraints. Instead of creating small and large (or different colored variations) versions of a graphic and In-Betweening, it is easier to transform an instance of the Symbol, and then In-Between Special. Second, the computer does the work. By setting up transformations properly, the computer will do what it does best - get the job done quickly. Beware: too drastic a change can cause trouble.

Color is complex and irritating.

It was only toward the end of the nineteenth century that color was standardized. Milton Bradley's school supply division was among the leaders in developing a consistent color system. Bradley's colorists decided on a blue and that was it. Matters are a bit more complicated with computers and their need to work with 256 colors. The long and short of it is: designers must choose a palette, construct a common palette, devise a special palette, or remap colors to an existing palette. It's all vey troublesome but not impossible.

All in all, exploring transforming media in Flash underscores the notion that Flash is but one tool in a designer's kit.

THE INTERACTIVE INTERFACE

Keep it simple.

Travelers to France who do not know French do not give speeches or have sophisticated discussions in French. They begin by memorizing a few simple sentences in order to ask for help finding the subway or a cathedral. Learn Lingo, first, by memorizing a basic vocabulary and a few common grammatical structures, second, by imitating and experimenting with others' use of Lingo, and last, by understanding and writing original Lingo.

Interactive Interface is all about feedback.

While graphic design is about how to lead the eye, interactive design is about how to lead the cursor. A user interface should be designed to clearly communicate the piece's options to the user. Like the world, the interactive interface should provide appropriate feedback about what is happening or has happened. When we turn on a light, for example, the light switch moves from the off (down) position to the on (up) position, a sound also accompanies the switch, and the light comes on. These three events provide feedback, confirming that the light has been turned on. When a user of a computer clicks on a button and expects something to occur but receives delayed or no feedback, he or she experiences a stress response. Sitting in front of the computer wondering if it is working can cause anxiety. Prompt and informative feedback is critical to interface design because it confirms the user's actions.

Multimedia interface means "more than one medium".

A user interface should be more than graphical. It should also be auditory. Sound is a visceral part of the world. The combination of sound and image creates an experience that is more complete and profound than either one. A user interface should also be animated because motion attracts the human eye. Multimedia, for the average user, is most similar to television or video games. Television keeps the screen subtly changing even when things are relatively static. Finally, an interface should be legible and readily understandable. Buttons with icons are effective once a user is familiar with a program, but legible text is more direct and readily understood by the new user.

Look at the world for interface ideas.

Buttons are just one form of interface. The world is full of things that all work in different ways. Finding a system in the world and using it as part of a GUI uses the concept of the metaphor. A department store might be represented by the metaphor of a catalog, an elevator, or a television, while the different departments might be represented by a set of pages, a particular floor, or a particular channel, respectively. Usually, a real-world metaphor takes advantage of people's familiarity with everyday objects or environments and their assumptions about how these objects operate or about an environment's geography.

FLOWCHARTING

There is no magic to flow charting.

There is no proper way. You will not be struck down by lightning if you use an “improper” symbol. Flow charting is a technique to help thinking. Use it in whatever way you want.

A place for everything and everything in its place.

One of the imperatives in programming is neatness. Quite often, when creating a multimedia project, you will need to look back at what you have done for reference. Keep your Scenes, Layers and Symbols well-named, tidy and organized. Group related media elements by using Folders in the Library window whenever possible. This will help you find them when you need them.

Design top down, build bottom up.

This is such an important axiom, it bears repeating again and again. The process of programming starts with conceptualization and concludes with implementation. The conceptualization is the invention of a multimedia piece. Start with an idea and expand on that idea until it is fleshed out. Implementation is the process of converting a set of plans into a concrete structure. Start by building the foundations and work up to the finishing touches.

The whole is the sum of its parts.

Build in chunks. Chunks are assembled into the whole. When creating the building blocks, focus on each as if it were the entire project. Don't be too anxious to see the finished product because it can't be rushed. The finished product relies on its building blocks and cannot be completed until all the components are finished. So, in order to feel the exhilaration of finishing a project more often, think of the whole project as the sum of lots of little projects.

INTEGRATION AND TESTING

Build, test, integrate, test.

It was either Ben Franklin or someone's mother who said, "An ounce of prevention is worth a pound of cure". In software development, the disease we are preventing is bugginess. For every two units of a project we put together, four units of bugs will be created. The best way to avoid the problem is to test as often as possible: after building each module, test it. When a tested module is ready to be integrated with another tested module, or set of modules, test the integration.

Know what you want, look for what you need.

Knowing what is needed is far more important than memorizing syntax. Unlike college students taking exams, programmers are free to consult manuals, on-line help, and colleagues. Multimedia developers must be like truffle-sniffing pigs. The most important skill to have for advanced development is the ability to smell out answers and solutions from a field of dirt. This is something that can only be developed through experience.

Abstraction rules.

The best programmers have an ability to manipulate abstractions. They can focus on a particular problem and break it into sub-problems ad nauseum. They can focus on the minutia of a problem to such a level that they can infer all possible outcomes to particular sub-problems. Some would say that they are, at heart, the best creative problem solvers around.

WORKING WITH DIGITAL VIDEO

Think broadly about movies.

Digital video can be used to animate buttons or other integrated screen elements. It can be used as a means to compress a series of unrelated stills. And digital video can be used as an audio track. Whatever its use, digital video is a versatile and flexible data type. By the same token, think expansively about Flash movies. A Flash movie need not fill the entire screen. It can be small, like a tool; or it can play within another Flash movie or Director movie. Finally, it need not look like a Flash movie at all but be an application (or web page) in its own right.

Small is even more beautiful.

“Small” covers a lot of territory when it describes digital video. One thing is true: digital video can get really big - but bigger does not always mean better. There are lots of ways to make digital video smaller. The height and width of the images, the frame rate, the compression technique, playback method, and the bit-depth of the images are all factors when it comes to shrinking digital video. The same principle, albeit with different considerations, holds true for shrinking your Flash movies. But the biggest space saving tool is the brain. Think like a magician and use illusion to get an effect with the least effort.

Compromise is a fine art.

In order to make digital movies and other varieties of Flash movies, and especially Shockwave animations, we must practice the fine art of compromise.