Where Does Flash Fit In?
by Phillip Kerman
Thunder Lizard's Web Design World Seattle, 23 July 2002

Where does Flash Fit in? Anywhere. In fact, it's not a matter of where and when Flash is appropriate but rather that its implementation is complete. If you're going to use it--use it right. There are situations where only Flash can achieve a particular result. Be aware of the following technical considerations to ensure success.

Plugin problems

Practically nonexistent. More than 95% of web users have the Flash 3 plugin which means you can use Flash to easily sense and upgrade their plugin. And if you add an "if you can't see this" link and you've got a quick-and-dirty fool proof method to reach everyone. For something a bit more automatic check out Macromedia's free "Flash Deployment Kit"
`http://www.macromedia.com/software/flash/download/deployment_kit/`
Naturally, those without the Flash plugin will need to download the approximately 300K player--but that's small (the Acrobat 5 plug-in, for example, is 8,600K!) and the install is automatic for most users. A very forthcoming disclosure of the Flash player's penetration (including the various version numbers) can be found at:
`http://www.macromedia.com/software/player_census/flashplayer/`

Accessibility

Macromedia has addressed this issue head-on with both an "Accessibility Kit" for Flash 5 as well as new features in the Flash Player 6 and Flash MX authoring tool. Existing features (in Flash 5) provide several ways that make content more accessible.

Built-in Accessibility Features Already Present:

Publish templates will automatically include text from your Flash movie in the HTML portion of your web page.

Keyboard equivalents can be assigned for any button eliminating the need for a mouse.

Automatic tabbing means that all onscreen buttons can be reached by tabbing (and the button highlights).

Synchronized Audio is available when using "stream event" sounds. This way closed captioning can be presented and will appear synchronized.

Magnification up to 2000% is available through the right-click (or control-click) Flash menu.

Accessibility Features in Flash MX (and Flash Player 6):

Flash text readable by screen reader devices (via Microsoft Active Accessibility with "Window Eyes").

Navigation within a Flash movie with targeted anchors. Because this will also allow the browser's **back button** to be used while remaining in a single Flash movie this is a general usability issue.

Accessibility (continued)

Think "equivalent":
When it comes down to it, you must think "equivalent content". This really has little to do with Flash--that is, the same rules apply to traditional static html... you must consider equivalent content.

For more on Accessibility:
`www.macromedia.com/macromedia/accessibility/`
`www.macromedia.com/macromedia/accessibility/features/flash/`
`www.macromedia.com/exchange/flash/`

Where To Use Flash / When to Pass on Flash
Different media types are effective for different messages. Consider the objectives that Flash is particularly well suited to resolve. And, understand Flash's limits.

People Respond to Different Stimuli:
Flash can target **visual** people. For example, people with good spatial relations can respond to a graphic map or 3D model. Additionally, some messages are simply best communicated by visual means. While other technologies employ static images, the added dimension that animation provides will often appeal to those attracted to **movement**. Again, some messages are particularly well suited to animation. Music changes a mood and sound effects add realism especially for **auditory** learners (for more, come to my presentation Friday "The Sound of Flash"). Flash can even appeal to **tactile** learners through conventional "drag and drop" interactions or proprietary "tactile mice".

Technical issues:
Building something really cool in Flash takes more skill than making a simple HTML page--but it has more value. There are other, less obvious, costs associated with Flash.

Flash requires a **specialized skill**. While Flash is very easy to use, to use it well requires a skill that you may not have access to. Your site may require maintenance--make sure you have the people to maintain it. (Additional investment early in a project can make the maintenance easy for anyone-- even those without Flash skills.)
The inherent anti-aliasing feature of Flash is great for making edges appear smooth. However, very **small typefaces** can become fuzzy. This figure exaggerates the issue: small anti-aliased text vs. small aliased text.
Flash is **proprietary**--but so what? So are GIF images for that matter. It's really not an issue because you can expect to encounter many proprietary technologies (or find it impossible to proceed). The good news is that Flash is well accepted and can even be licensed for zero cost.
Other technical issues are discussed under "Flash & HTML Together", below.

Usability

Poor usability has been around for a long time--it has only become a subject of conversation for Flash because along with Flash's ease-of-use has come those who have created examples of bad usability. Today you must be extra careful with Flash because your content is suspect--be sure to consider usability.

Traditional Usability Concepts Applied to Flash:

Leverage off standards

Don Norman suggests that making the user learn a new standard is fine--but don't make them learn something new for every case. Take advantage of those that went before you and *use* standards.

Avoid "chart junk"

Edward Tufte refers to frivolous graphics as "chart junk". Basically, screen space is very limited. (See "if it's not adding anything…" below.)

Multiple routes to the same destination is okay

Some people, in their zeal to maintain consistency, will argue having two buttons that do the same thing is redundant. Not only do people learn differently, but people like to select their own patterns to follow. For example, you might be a "menu person" or a "toolbar person".

Matching labels

Consistency is important when labeling interface components (such as the buttons). You're effectively trying to teach the user how things work. For example, don't make your "main" button change from "main" to "home"--pick one name and stick with it.

1 of 2 Dilemma

A very common issue arises when graphically representing a "selected" state for a button if there are only two buttons present. No matter how you treat it, in the end you'll simply have one button with the highlight and one without. It is particularly difficult for the user to know whether the highlight or absence indicates "selected". Be sure you do something extra to make the selected item stand out.

If it's not adding anything then it's distracting

Force yourself to justify every additional graphic, sound, or animation. Everything should have a purpose. Not only do gratuitous effects become tedious, but can make for larger files (that download slower).

Repetition can be okay

When you *do* repeat elements try to introduce variations. For example, if the same block of text is repeated in several places within your site put the repeated part after any new text. That way, users who stop reading when they see the same words won't miss anything.

Avoid hard-and-fast rules (e.g. 7 +/- 2)

Of course all these "rules" are suggested guidelines. Rules such as "seven items plus or minus 2" can be misinterpreted and inappropriate to your situation. Learn the traditional rules so that when you break them you can justify your actions.

Traditional Usability Concepts (continued)

Tiny is bad

Layouts that include tiny buttons may be visually appealing but hard to click--give your users a break. In such cases, make large "hit" areas so the buttons are easy to click. (This is particularly easy with Flash buttons.)

If you can't figure it out, how can they?

If you <u>ever</u> find yourself (or anyone on your team) misunderstanding an interface component, treat this as a sign that you must fix it! For example, say you keeping clicking the "Back to Main" button when you're intending to only go back one page... if it happens to you even once realize it will likely happen to your users too.

Flash & HTML Together:

Flash can be embedded in HTML in any of the same ways that images can. For example, **tables**, **frames**, and **popup** windows are all possible.

Through **JavaScript**, Flash can "talk" to other Flash movies embedded in a web page. Technical considerations include ensuring target movies have downloaded and that the user has JavaScript enabled.

Additionally, Flash can talk to servers via **GET** and **POST** by sending strings of **XML** structured data. And, with a custom Java server, Flash can establish persistent connections using XML Sockets.

"Applications" not "Pages":

Flash (with connected MX products) is now suited to provide full fledged web-applications. The breaks the old metaphor of pages.

Summary:

Flash is a **standard** that's feasible for general audiences.
Accessibility is an emerging issue, unrelated to Flash, although there are *some* current solutions inside Flash and more to come.
Usability is no more or less important with Flash than any other communication.
Flash **fits seamlessly** inside "traditional" HTML.

References:

The Design of Everyday Things
By Donald Norman Published by Doubleday Books (ISBN: 0385267746)

3 Books by Edward Tufte / Published by Graphics Press
The Visual Display of Quantitative Information (ISBN: 0-9613921-0-X)
Envisioning Information (ISBN: 0-9613921-1-8)
Visual Explanations: Images and Quantities, Evidence and Narrative (ISBN: 0-9613921-2-6)

My books (**www.phillipkerman.com/books**):
ActionScripting in Flash MX Published by New Riders (ISBN: 0735712956)
Sams Teach Yourself Flash MX in 24 Hours Published by Sams (ISBN: 0-672-32371-0)