



## **When Flash Fails**

“Progressive enhancement” and its  
goateed evil twin “progressive failure”

**APCO**  
worldwide®

January 2010

## When Flash Fails

### “Progressive enhancement” and its goateed evil twin “progressive failure”

We all know the value of being prepared. When a project doesn't go precisely as planned (and what project does?), it's always best to anticipate some likely scenarios and prepare solutions in advance. We must understand possible points of failure so that we can plan appropriately and make sure that we can achieve our goals regardless of the challenges. Especially the challenges we are able to predict.

Why would we approach a Web project any other way?

People interact with the Web in many different ways, and yet we expect one Web page to meet the needs and peculiarities of different computers, settings, browsers, service providers and users. Worse, owing to cautious corporate policies, business Web browsers restrict access to some features or may be years out of date – consider the poor sap running Internet Explorer (IE) 6, a browser that was released eight years ago. One might think that IE6 is long gone, but [as recently as October 2009](#), 23 percent of people on the Internet were still using it.

So what does this have to do with failure? Everything. People expect dynamic content – content that moves and updates as they watch, content that links to their social networks, that lives and reacts, instead of just sitting static on the page. The trouble is presenting this living content in a way that can be viewed by the many browsers and technologies your audience may be using at work, at home and on the go.

For years the answer has simply been to use Flash. Adobe's Flash plugin has provided a rich set of tools for creating interactive Web content, as well as a reliable compatibility baseline. This means that a Flash feature running in IE6 on Windows will behave the same as in Safari 4 on Mac OSX, Firefox 3.5 on Linux, and so on. Furthermore, the Flash programming interfaces are well known, well documented, and best of all, the same regardless of the browser hosting the plugin. Finally (and justifyingly), Flash allows for fancy client-pleasing graphical effects such as 2D & 3D rotations which aren't generally available in plain Jane HTML.

### So what's wrong with Flash?

**People with disabilities can't use Flash:** Blind people and people with certain motor function disabilities cannot interact with a Flash applet at all. Companies have been sued over this. A well-know retail chain for example, was sued for making their Web site inaccessible to blind visitors. While this wasn't a Flash issue, it represents the same negligence to accessibility.

**Search engines don't see Flash:** Content presented in Flash cannot be indexed by search engines or searched using a Web browser's text search functionality, making it harder for your audience to find your content.

**Flash doesn't print well:** Content presented in Flash rarely can be printed effectively. To Adobe's credit, they provide mechanisms for authors of Flash content to make it printable, but those programming interfaces are arcane, and hard to use correctly – hard enough to use that this author has only seen it done correctly by Adobe themselves, and never in the wild.

**Popular devices don't support Flash:** Finally, to add insult to injury, a number of platforms simply do not support Flash and are unlikely ever to do so – platforms such as the iPhone and the Blackberry, which are hardly niche devices. Apple has made it clear they do not intend – ever – to support the Flash plugin on the iPhone's Web browser.

So in short, Flash is a magnificent tool, which fails completely for a small percentage of visitors. If the phrase "small percentage of visitors" seems benign, consider rephrasing it to "a small percentage of possible customers" or "key stakeholders." Thus we have to consider how to mitigate this failure – we don't ever want to drive off our audience, nor do we want to alienate the disabled.

Given a medium that must occasionally fail, we need to address how it fails.

### **How Flash Fails**

When Flash fails, it pretty much fails wholesale: think of a helicopter running out of fuel and dropping like a stone. Flash works correctly, or not at all. This means that your content will be totally unviewable (and your message completely lost) for the segment of your audience who cannot view Flash.

A well-designed site should allow your content to be read even if it is not rendered correctly.

There are options. Other standard mechanisms of presentation on the Web can stand independently. Like a tower supported by buttresses, the tower is stronger when intact, but will not fall apart if one buttress fails. If one part breaks, a site may look ugly or lose its interactivity, but the content – and your message – will remain legible.

### **Progressive Enhancement and Failure**

This is the crux of what we call "progressive enhancement," and its counterpart "progressive failure." The spirit of progressive enhancement is to layer interactivity and fancy design on top of a functional if bare-boned core. Anybody with even the

rudest and most obsolete Web browser should be able to see the core and read your content. Blackberries, iPhones, screen readers for the blind, and even a hoary 1990 Radio Shack Tandy computer running a text-only Lynx browser should be able to read your copy and navigate your site. But visitors with state of the art browsers with greater functionality will be able to properly enjoy the design and interactivity.

So your site is visible and fancy with a good modern browser, but it is also usable by the blind, by people with mobile browsers, or corporate-dictated ancient browsers such as IE6. This is progressive failure.

Where does this leave us? We need to reconsider where Flash fits in. Flash was designed to be a format for interactive animation, and it still excels in this regard. It remains a fantastic platform for games, cartoons and advertisements. However, while we do use Flash to these ends at APCO, we also continue to use it for slideshows, “virtual tours” and other circumstances where we’d be better served by Web browsers’ native capabilities. We have a choice now, unlike 10 or even five years ago, and it would be best for our clients (and their clients) if we take these modern approaches.

It pays to be prepared. By weighing your needs, your audience and the tools at their disposal, your online project manager can make a recommendation that takes into account these principles of progressive enhancement and progressive failure. A bit of thoughtful planning and prioritization can help to ensure that your message is delivered intact to the people who need to see it, while offering the styling and interactive features you’ve come to expect to those who can handle it.

---

What we're reading:

[A List Apart](#)

Well-written articles on the zen of web design/programming/writing

[Chrome Experiments](#)

Mind-blowing demos of what modern browsers are capable of

[Ajaxian](#) & [InsideRIA](#)

News on and articles about developing tech and techniques on the browser side

*Shamyl Zakariya is an art director with StudioAPCO, APCO Worldwide's award-winning team of producers and designers with experience ranging from film and video production to innovative print publishing and the latest in interactive and digital design. He can be reached at [SZakariy@apcoworldwide.com](mailto:SZakariy@apcoworldwide.com).*