Viewing Websites in Different Web Browsers

It never fails: No matter which browser you use, and how well you keep up with its updates, sooner or later you end up finding a website that doesn't quite work the right way. Even more frustratingly, bugs and defects always seem to crop up more often on sites that tend to push the limits of what can be done with a browser, with activities like printing, interactivity, and complex animations frequent targets of malfunctions and misfirings.

It's easy to blame Web developers for being lazy and allowing these problems to occur, but the reality is that the Web has become a complex, well, web of technologies and competing interests that are, sometimes, at odds with the goal of writing software that works across as many platforms as possible.

A fuzzy standard

Web pages are programmed primarily using three different languages: HTML, used to define the structure of documents; CSS, which determines the way they look; and JavaScript, which is used to drive interactive features.

Of these, the first two are "managed" by a non-profit organization called the World Wide Web Consortium, or W3C, that is responsible for defining the standards that—in theory—browser manufacturers and developers must follow in order for their products to be compatible with each other.

In practice, however, Web standards tend to trail de facto innovations introduced by the individual browsers, usually in response to the business needs of their makers. For example, Apple introduced a number of new features when it launched Safari for iOS, all aimed at making the company's flagship browser work well with its new touch interface; in typical Cupertino fashion, this was done in secrecy, leaving those responsible for the corresponding standards to catch up after the fact.

Thus, the specification behind the fifth incarnation of HTML is not expected to be finalized until 2014, and won't become a "Recommendation"—the status accorded by the W3C to official standards—until 2016. Yet HTML5, as it is commonly called, is already widely used by all major browsers and many, many websites, although support for HTML5 is implemented in different ways by each browser vendor.

A web of complexity

Internally, every browser uses a component called a rendering engine, which is responsible for transforming source code into the webpage you see. Each major vendor has developed its own incarnation, with the exception of Google and Apple, which have based their Chrome and Safari browsers, respectively, on WebKit, an open source rendering engine to which both companies contribute.

There are sometimes profound different differences in the way each engine decides to interpret and render a particular webpage. In fact, even Chrome and Safari, which are based on the same underlying software, often exhibit completely different behavior towards the same piece of code. And, to make matters even worse, it's not unusual for different versions of the same engine to be at odds with each other, so that, for example, Safari on OS X 10.6 may well render webpages in a different way from its 10.8 counterpart.

Of rocks and hard places

In between these competing business needs and shaky standards, Web developers are tasked with the job of coding sites that takes advantage of as many advanced browser features as practical while reaching the widest possible audience—a job that can be downright impossible. Once written, sites must then be tested with as many different browser and operating system combinations as possible—another potentially monumental task that is time consuming and has a knack for resisting automation and one that can easily be, once it reaches a certain complexity, beyond the budgetary capabilities of many development shops.

Thus, most developers seek to reach a compromise based on the relative popularity of the browsers that their visitors use. A site that tries to cater to as wide an audience as possible tends to use fewer cutting-edge features, relying instead on time-proven techniques that are more likely to work even with old operating systems and browsing platforms. A site that has a more specialized target market, on the other hand, will tend to narrow down support to a handful of specific browsers so that it can take advantage of more recent advances in Web technologies. For example, though it does work with the Windows-only Microsoft Internet Explorer browser, Macworld.com could probably get away without supporting that particular option, while ignoring Safari would likely be a less optimal choice.

Keeping a backup handy

Inevitably, however, small issues have a way of falling through the cracks here and there: Printing might work flawlessly in Chrome, but not in Safari, and maybe only on a specific version of OS X. Some sites still insist on only working well with Internet Explorer, despite the popularity of devices—like the iPhone and the iPad—that only allow their users to browse the Web with WebKit-based browsers. What is one to do in these cases?

On desktop operating systems, the answer is as simple as it is pithy: Use a different browser. In my experience, while small cosmetic problems are fairly commonplace, websites that refuse to work at all with a modern browser like Safari or Chrome are relatively rare. When I come across them, however, I simply switch to Mozilla's Firefox browser as a backup.

On iOS devices, this problem is much harder to tackle, because Apple only allows third-party browsers to use the same version of Webkit that powers the mobile version of Safari. Still, in a pinch you can try an app like Opera Mini, whose servers pre-process and compress Web pages to improve their compatibility with iOS's built-in rendering engine.

In the end, while its results can sometimes be annoying, the competition between browser manufacturers has helped the Web evolve from the ugly pages of the 1990s into a technology that, today, encompasses everything from high-definition video to realtime 3D graphics. All in all, the occasional hiccup seems like a small price to pay for an open and vibrant Internet experience in which blinking text has been replaced by much more impressive memes. - *Marco Tabini*

PITE creative