

Google turbo-charging the back button with Chrome's new "back/forward cache"

Enlarge / Now that's some shiny chrome.

reader comments

50 with 43 posters participating

Share this story

Google is developing a new cache for Chrome (via CNET) that should make some page loads extremely fast. The only catch? They'll have to be pages you've already seen and are revisiting after hitting the browser's back button.

Chrome already caches the files that make up a page, so revisiting a page in most circumstances shouldn't force the browser to retrieve the images, JavaScripts, and CSS that are used to build the page. But currently, the browser has to re-parse the HTML and re-build the page's programmatic representation, uncompress the images, re-execute all the JavaScript, reapply all the stylesheets, and so on. It's just the networking step that gets skipped.

The new bfcache (for "back/forward cache") changes that: it lets the browser capture the entire state of a running page—including scripts that are in the middle of execution, the rendered images, and even the scroll position—and reload that state later. With bfcache, rather than having to reload the page from scratch, the page will look as if it was paused when you click a link to a new page and subsequently resumed

when you hit back.

Unlike file caches—which can accelerate the loading even of new pages and can span browsing sessions—the bfcache will only accelerate pages you’ve already recently visited. As the name implies, it’s only for navigation by hitting the back and forward buttons to revisit your history. But Google says that such visits represent a significant part of typical browsing sessions, with some 10 percent of pages on the desktop and 19 percent of pages on mobile revisited in this way.

Safari and Firefox have been using a similar kind of caching for years. Safari calls it the Page Cache and has had it since 2002, while Firefox’s is called the BFCache, and it’s been present in some form since Firefox 1.5 in 2005. While Google’s Blink engine has common heritage with the WebKit engine of Apple’s Safari, Google has chosen not to use WebKit’s counterpart to the bfcache, saying that it’s incompatible with the multiprocess model that Chrome uses.

Details, details

The devil with this kind of cache is in the details. Safari didn’t start using the Page Cache for pages with frames until 2009, and it didn’t cache sites served over HTTPS until 2012. Pages that use scripts to detect when they’re being navigated away from can’t be cached, and Firefox and Safari both include alternative facilities so that scripts can tell when their page is being suspended or resumed. The new cache should have some improvement in battery life, as resuming a suspended page will be a lot less intensive than rebuilding the page. But the bfcache also threatens to increase Chrome’s memory usage, something that’s already a frequent concern with the browser.

The work to enable this in Chrome is going to be substantial; a number of different parts of the browser need to coordinate to ensure that every part of a page, including background scripting, is frozen correctly. Accordingly, the feature is

going to be developed this year, but i isn't likely to be found in stable builds of Chrome until 2020.

<https://arstechnica.com/?p=1465915>