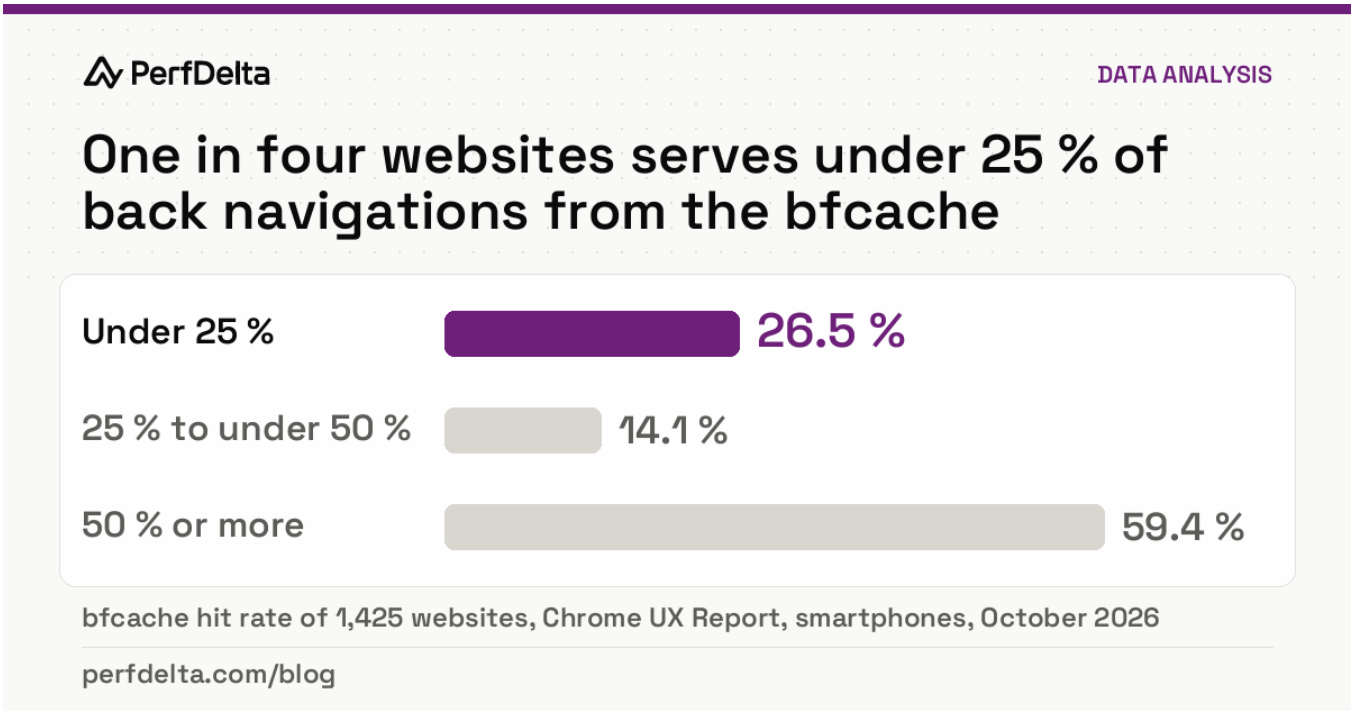
bfcache: One in Four Websites Reloads Most Pages After the Back Button

Published November 3, 2026 by Christian Schreiber



When a visitor taps the back button, the previous page can appear instantly. Browsers keep recently visited pages in memory, in the back/forward cache (bfcache), and restore them without a single network request. That only works if the page allows it. We looked at the Chrome field data of 1,425 websites: on one in four, fewer than a quarter of back and forward navigations came from the bfcache. All other back navigations loaded the page again.

For these websites, a fast return to the previous page is the exception. The good news: the most common blockers can be removed with small changes, and none of them require a faster server.

The bfcache turns back navigations into instant page loads

Without the bfcache, the back button works like a new visit. The browser requests the HTML again, loads scripts and stylesheets, runs JavaScript and builds the page. With the bfcache, the browser keeps a complete snapshot of the page in memory, including the scroll position and the state of forms and scripts. Going back simply puts that snapshot on screen again.

Think of a book you put down with a bookmark in it. With the bfcache, you open it at the right page. Without it, you start again at the cover and leaf through to where you were. Google describes the

bfcache as one of the most effective performance optimizations for real users in its guide Back/forward cache (web.dev, last updated October 2026).

Back and forward navigations are about one in nine page views

History navigations are not a rare edge case. On the median website in our sample, 11.4 percent of all page views on smartphones were back or forward navigations. That is roughly every ninth page view. Visitors jump back from a product page to the category, from an article to the overview or from a checkout step to the cart, and expect the previous page to be there immediately.

One in four websites misses the bfcache on most back navigations

The bfcache hit rate is the share of back and forward navigations that Chrome served from the bfcache. We looked at all websites in our sample where at least two percent of page views were back or forward navigations:

bfcache hit rate	Share of websites
Under 25 %	26.5 %
25 % to under 50 %	14.1 %
50 % or more	59.4 %

The median website served 59 percent of its history navigations from the bfcache. The best quarter reached 77 percent or more, while the weakest quarter stayed below 19 percent. A hit rate of 100 percent is not realistic, since browsers also evict pages from memory when it runs low or when a page has been in the background for a long time. A rate far below the median, however, almost always points to a blocker on the website itself.

Cache-Control: no-store is the most common bfcache blocker

A browser only keeps a page in the bfcache if it can safely restore it later. Some instructions and features in a page rule that out. The most common ones:

- **`Cache-Control: no-store` on the HTML.** Many content management systems and shop platforms send this header on every page, often only because the page could contain a shopping cart or a login state. According to Chrome, this header was the largest single reason for missed bfcache restores, affecting about 17 percent of history navigations on mobile (Enabling bfcache for Cache-Control: no-store, Chrome for Developers, September 2025). Since spring 2025, Chrome allows such pages into the bfcache under certain conditions, but other browsers still exclude them, and the cleanest solution is to send `no-store` only on pages that really contain personal data.

- **`unload` event handlers.** Scripts that run when a page is left, often from older analytics or chat tools, used to block the bfcache in Chrome. Chrome has now completed the deprecation of the unload event, so these handlers no longer fire by default (Deprecating the unload event, Chrome for Developers, updated October 2026). Websites that explicitly re-enable unload, and other browsers, still benefit from switching to the `pagehide` event.
- **Open connections.** Pages with an open IndexedDB connection or WebRTC connection when the visitor leaves cannot be stored safely. Open WebSockets no longer block the bfcache in Chrome (since version 149) and Safari, but still do in Firefox. Closing these connections on `pagehide` and reopening them on `pageshow` solves this.
- **References between windows.** If a page opens a window with `window.open()` and keeps controlling it, neither page is eligible for the bfcache. Links with `target="_blank"` no longer create such a reference in modern browsers, since `rel="noopener"` is now the default there.

Testing bfcache eligibility takes two minutes

Chrome DevTools has its own test: in the Application panel under Back/forward cache, a click on Test back/forward cache navigates away and back and lists every reason that prevented the restore (Test back/forward cache, Chrome for Developers). Lighthouse runs the same check in its diagnostic Page prevented back/forward cache restoration, which does not affect the performance score. Both test a single page in the lab. Whether the fix works for real visitors only shows in the field data a few weeks later.

When the bfcache does work, it also helps the Core Web Vitals. A page restored from the bfcache appears almost immediately, and the Chrome UX Report counts each restore as a separate page visit, so fast restores improve the field values. The Core Web Vitals guide explains how field data is collected.

Check the bfcache hit rate of your website

PerfDelta shows the bfcache hit rate of your domain from real smartphone visits in every report, next to the share of back and forward navigations and the values of your competitors. A week-by-week history shows whether a fix is taking effect. Start with a free page speed test or compare up to five websites directly in the Core Web Vitals benchmark.

Method and limits of this analysis

- **Data:** 2,003 smartphone records from the Chrome UX Report, each origin counted once, data period up to October 26, 2026, retrieved by PerfDelta. Each record covers a rolling 28-day window of real Chrome visits.
- **Selection:** the 1,425 websites on which back and forward navigations made up at least two percent of page views. Below that, the hit rate rests on too few navigations.

- **Hit rate:** navigations of the type back_forward_cache divided by all back and forward navigations, with and without the bfcache.
- **Chrome only:** the Chrome UX Report contains no data from Safari or Firefox, which have their own bfcache implementations.
- **Sample:** the websites come from PerfDelta tests and studies and lean towards larger brands. The results are not representative of the entire web. More figures from our measurements are in the website speed statistics.