

# History API: Scroll Restoration



By Paul Lewis

Paul is a Design and Perf Advocate

Using the History API to manage your URLs is awesome and, as it happens, a crucial feature of good web apps. One of its downsides, however, is that scroll positions are stored and then, more importantly, *restored* whenever you traverse the history. This often means unsightly jumps as the scroll position changes automatically, and especially so if your app does transitions, or changes the contents of the page in any way. Ultimately this leads to an horrible user experience.

To make matters even worse there's very little you can do about it: Chrome triggers a `popState` event before the `scroll` event, which means you can read the current scroll position in `popState` and then reverse it in the `scroll` event handler with `window.scrollTo` (Ewww, but at least it works!). Firefox, however, triggers the `scroll` event *before* `popState`, so you have no idea what the old scroll value was in order to restore it. Bah!

The good news is, however, that there's a potential fix: **`history.scrollRestoration`**. It takes two string values: `auto`, which keeps everything as it is today (and is its default value), and `manual`, which means that you as the developer will take ownership of any scroll changes that may be required when a user traverses the app's history. If you need to, you can keep track of the scroll position as you push history entries with `history.pushState()`.

Here's a quick video showing the difference between `auto` and `manual` values:

The feature is new and experimental (though totally awesome), so be sure to check it's available before using it:

```
if ('scrollRestoration' in history) {  
  // Back off, browser, I got this...
```



```
    history.scrollRestoration = 'manual';  
}
```

You will find `history.scrollRestoration` in Chrome 46 onwards, and you can [find its spec here](#).

Don't forget to leave us feedback, and let other vendors know if you want them to support `scrollRestoration`, too.

---

*Except as otherwise noted, the content of this page is licensed under the [Creative Commons Attribution 3.0 License](#), and code samples are licensed under the [Apache 2.0 License](#). For details, see our [Site Policies](#). Java is a registered trademark of Oracle and/or its affiliates.*

*Last updated July 2, 2018.*