

# Deprecations and Removals in Chrome 59



By Joseph Medley

Technical Writer

In nearly every version of Chrome, we see a significant number of updates and improvements to the product, its performance, and also capabilities of the Web Platform. This article describes the deprecations and removals in Chrome 59, which is in beta as of April 27. This list is subject to change at any time.

## Temporarily disable `navigator.sendBeacon()` for some blobs

The `navigator.sendBeacon()` function has been available since Chrome 39. As originally implemented, the function's `data` argument could contain any arbitrary blob whose type is not CORS-safelisted. We believe this is a potential security threat, though no one has yet tried to exploit it. Because we do NOT have a reasonable immediate fix for it, temporarily, `sendBeacon()` can no longer be invocable on blobs whose type is NOT CORS-safelisted.

Although this change was implemented for Chrome 60, it is has since been merged back to Chrome 59.

### Chromium Bug

## Remove features from WebVR that are not in the revised spec

The current implementation of WebVR, originally implemented in Chrome 52, contained several methods and properties that will not be in the final spec. Deprecation messages were added for these features for the Origin Trial that started in Chrome 56. These features and are now being removed. They include:

- `VRDisplay.getPose()`
- `VRDisplay.resetPose()`
- `VRDisplay.isConnected`
- `VRDisplayCapabilities.hasOrientation`
- `VREyeParameters.fieldOfView`

[Intent to Experiment](#) | [Chromestatus Tracker](#) | [Chromium Bug](#) | [Origin Trial Results so Far](#)

## Remove FileReaderSync from service workers

The Service Worker spec has always had the (non-normative) note that "any type of synchronous requests must not be initiated inside of a service worker", to avoid blocking the service worker (as blocking the service worker would block all network requests from controlled pages). However synchronous APIs such as `FileReaderSync` were still available in service workers. `FileReaderSync` was deprecated in Chrome 57. It is removed in Chrome 59.

[Intent to Deprecate](#) | [Chromestatus Tracker](#) | [Chromium Bug](#)

## Remove non-standard DeviceOrientation Event initialization functions

For some time now there's been a general trend in browser APIs away from initialization functions and toward object constructors. The most recent version of the [DeviceOrientation Event Specification](#) follows this trend by requiring constructors for both [DeviceOrientationEvent](#) and [DeviceMotionEvent](#).

Since Chrome is [enabling these constructors by default](#) in Chrome 59 the legacy initialization functions, `initDeviceMotionEvent()` and `initDeviceOrientationEvent()` are also removed. Edge has deprecated the initialization functions and Firefox has already shipped the constructors.

[Intent to Remove](#) | [Chromium Bug](#)

## Remove "on-demand" value for hover/any-hover media queries

The "on-demand" value for hover/any-hover media queries was removed from the spec about a year ago. Consequently, these media queries are removed in Chrome 59.

[Intent to Remove](#) | [Chromestatus Tracker](#) | [Chromium Bug](#)

## Remove MediaStreamTrack.remote

In Chrome 48 the `MediaStreamTrack.remote` property was added in support of the [Media Capture and Streams API](#) with the goal of allowing JavaScript to know whether a WebRTC `MediaStreamTrack` is from a remote source or a local one.

Since that time, this property has been removed from the spec. As of Chrome 59, it is no longer supported.

[Chromium Bug](#)

## Remove support creating `ProgressEvent` with `document.createEvent()`

Earlier versions of the DOM spec required implementation of `document.createEvent("ProgressEvent")`. However usage was always low and support has already been removed from [Gecko](#) and [Webkit](#). The event itself was [removed from the spec](#) in March of this year.

To conform with the platform and most recent spec, `ProgressEvent` is now removed from Chrome.

[Chromium Bug](#)

## Remove `SVGTests.requiredFeatures`

In the first version of the SVG spec, an application could call `DOMImplementation.hasFeature` to verify that a particular SVG interface is supported. Many SVG elements contained a `requiredFeatures` attribute that returned the same information.

In SVG2 `DOMImplementation.hasFeature` property always returns true. Consequently `requiredFeatures` no longer does anything useful. Because it was [removed from the spec](#) it was deprecated in Chrome 54 and has now been removed.

[Intent to Remove](#) | [Chromestatus Tracker](#) | [Chromium Bug](#)

## Deprecation policy

To keep the platform healthy, we sometimes remove APIs from the Web Platform which have run their course. There can be many reasons why we would remove an API, such as:

- They are superseded by newer APIs.
- They are updated to reflect changes to specifications to bring alignment and consistency with other browsers.
- They are early experiments that never came to fruition in other browsers and thus can increase the burden of support for web developers.

Some of these changes will have an effect on a very small number of sites. To mitigate issues ahead of time, we try to give developers advanced notice so they can make the required changes to keep their sites running.

Chrome currently has a [process for deprecations and removals of API's](#), essentially:

- Announce on the [blink-dev](#) mailing list.
- Set warnings and give time scales in the Chrome DevTools Console when usage is detected on the page.
- Wait, monitor, and then remove the feature as usage drops.

You can find a list of all deprecated features on [chromestatus.com](#) using the [deprecated filter](#) and removed features by applying the [removed filter](#). We will also try to summarize some of the changes, reasoning, and migration paths in these posts.

---

*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.*