

Uses An Excessive DOM Size

Overview

A large DOM tree can harm your page performance in multiple ways.

- Network efficiency and load performance. If your server ships a large DOM tree, you may be shipping lots of unnecessary bytes. This can also slow down page load time, because the browser may be parsing lots of nodes that aren't even displayed above-the-fold.
- Runtime performance. As users and scripts interact with your page, the browser must constantly re-compute the position and styling of nodes. A large DOM tree in combination with complicated style rules can severely slow down rendering.
- Memory performance. If you use general query selectors such as `document.querySelectorAll('li')` you may be unknowingly storing references to a very large number of nodes, which can overwhelm the memory capabilities of your users' devices.

Recommendations

An optimal DOM tree:

- Has less than 1500 nodes total.
- Has a maximum depth of 32 nodes.
- Has no parent node with more than 60 child nodes.


In general, look for ways to create DOM nodes only when needed, and destroy them when no longer needed.

If your server ships a large DOM tree, try loading your page and manually noting which nodes are displayed. Perhaps you can remove the undisplayed nodes from the loaded document, and only create them after a user gesture, such as a scroll or a button click.

If you create DOM nodes at runtime, [Subtree Modification DOM Change Breakpoints](#) can help you pinpoint when nodes get created.

If you can't avoid a large DOM tree, another approach for improving rendering performance is simplifying your CSS selectors. See [Reduce The Scope And Complexity Of Style](#)

[Calculations.](#)

[Audit source](#) 

Feedback

Was this page helpful?

YES

NO

Great! Thank you for the feedback.

Sorry to hear that. Please [open an issue](#) and tell us how we can improve.

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 24, 2018.