

async vs defer attributes

Published 26 February 2014

[HTML](#) [JavaScript](#)

The **async** and **defer** attributes for the `<script>` element have great support now, so it's time to learn exactly what they do!

Ecommerce users make 1000s of decisions. With AI, there is a better way to predict them [Learn More](#)

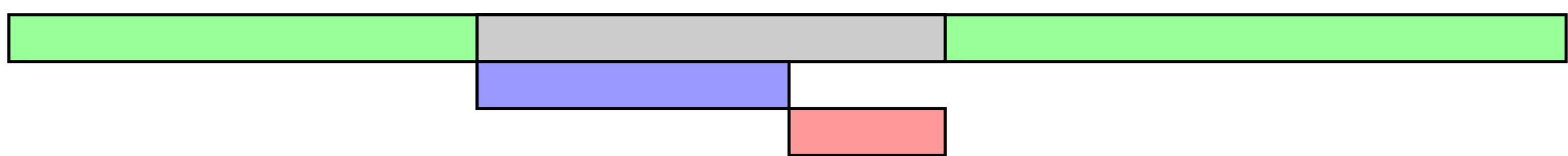
Ads by EthicalAds

Legend

- HTML parsing
- HTML parsing paused
- Script download
- Script execution

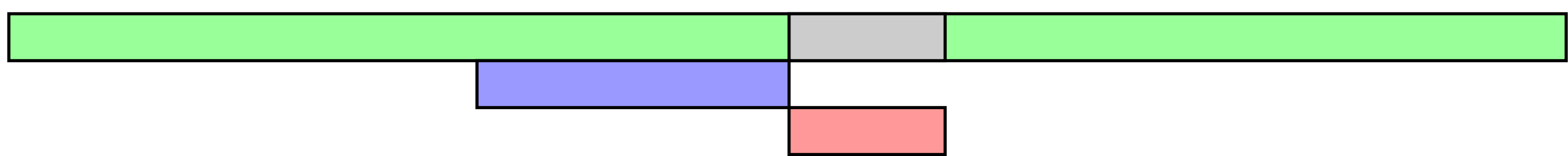
`<script>`

Let's start by defining what `<script>` without any attributes does. The HTML file will be parsed until the script file is hit, at that point parsing will stop and a request will be made to fetch the file (if it's external). The script will then be executed before parsing is resumed.



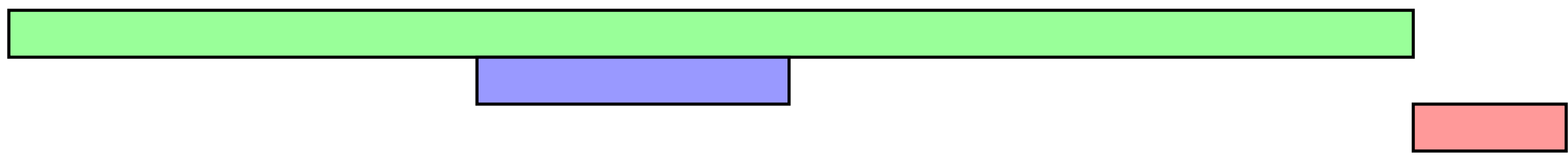
`<script async>`

async downloads the file during HTML parsing and will pause the HTML parser to execute it when it has finished downloading.



`<script defer>`

defer downloads the file during HTML parsing and will only execute it after the parser has completed. **defer** scripts are also guaranteed to execute in the order that they appear in the document.



When should I use what?

Typically you want to use **async** where possible, then **defer** then no attribute. Here are some general rules to follow:

- If the script is modular and does not rely on any scripts then use **async**.
- If the script relies upon or is relied upon by another script then use **defer**.
- If the script is small and is relied upon by an **async** script then use an inline **script** with no attributes placed *above* the **async** scripts.

Support

IE9 and below have some pretty bad bugs in their implementation of **defer** such that the execution order isn't guaranteed. If you need to support \leq IE9 I recommend not using **defer** at all and include your scripts with no attribute if the execution order matters. [Read the specifics here](#).

Share this page



More posts tagged [JavaScript](#)

- [AVL tree](#)
- [Binary heap](#)
- [Binary search tree](#)
- [Binomial heap](#)
- [Fibonacci heap](#)
- [How to fix npm slowing down terminal initialisation](#)
- [Migrating xterm.js from TSLint to ESLint](#)
- [Red-black tree](#)
- [Splay tree](#)
- [Using the Intersection Observer web API to improve performance](#)

Follow me

