“Tight Mode”: How Browsers REALLY Load Web Pages

Robin Marx
@programmingart

Akamai

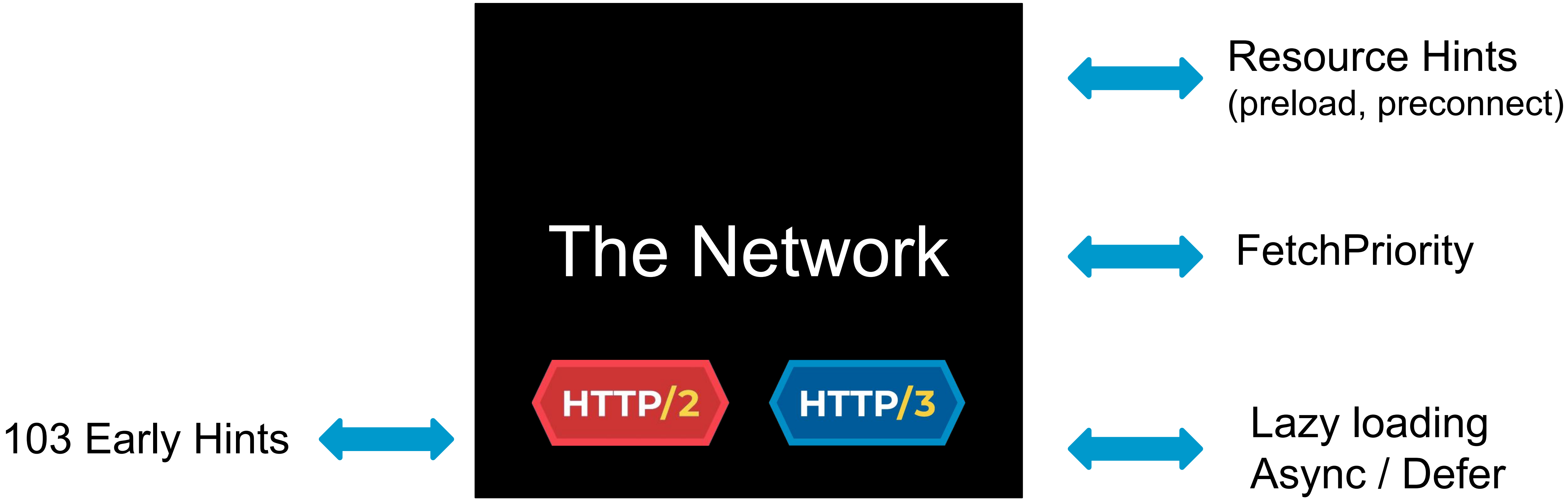
“Tight Mode”: How Browsers REALLY Load Web Pages

Akamai

Robin Marx
@programmingart

Barry Pollard
(Google Dev Advocate)

Tim Vereecke
(My colleague, giving talk next door)



The Network

HTTP/2

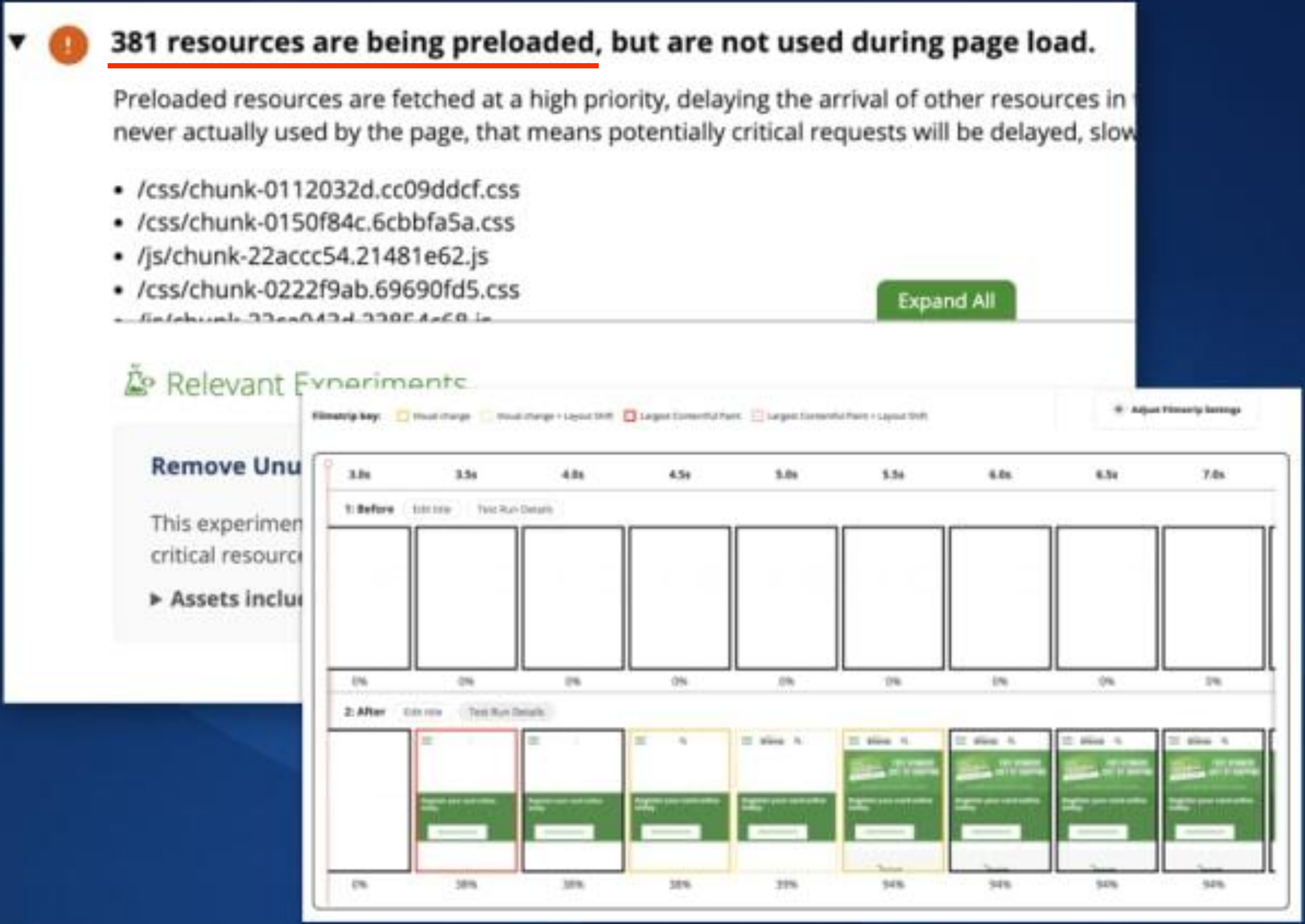
HTTP/3

Resource Hints
(preload, preconnect)

FetchPriority

Lazy loading
Async / Defer

103 Early Hints



381 resources are being preloaded, but are not used during page load.

Preloaded resources are fetched at a high priority, delaying the arrival of other resources in
never actually used by the page, that means potentially critical requests will be delayed, slow

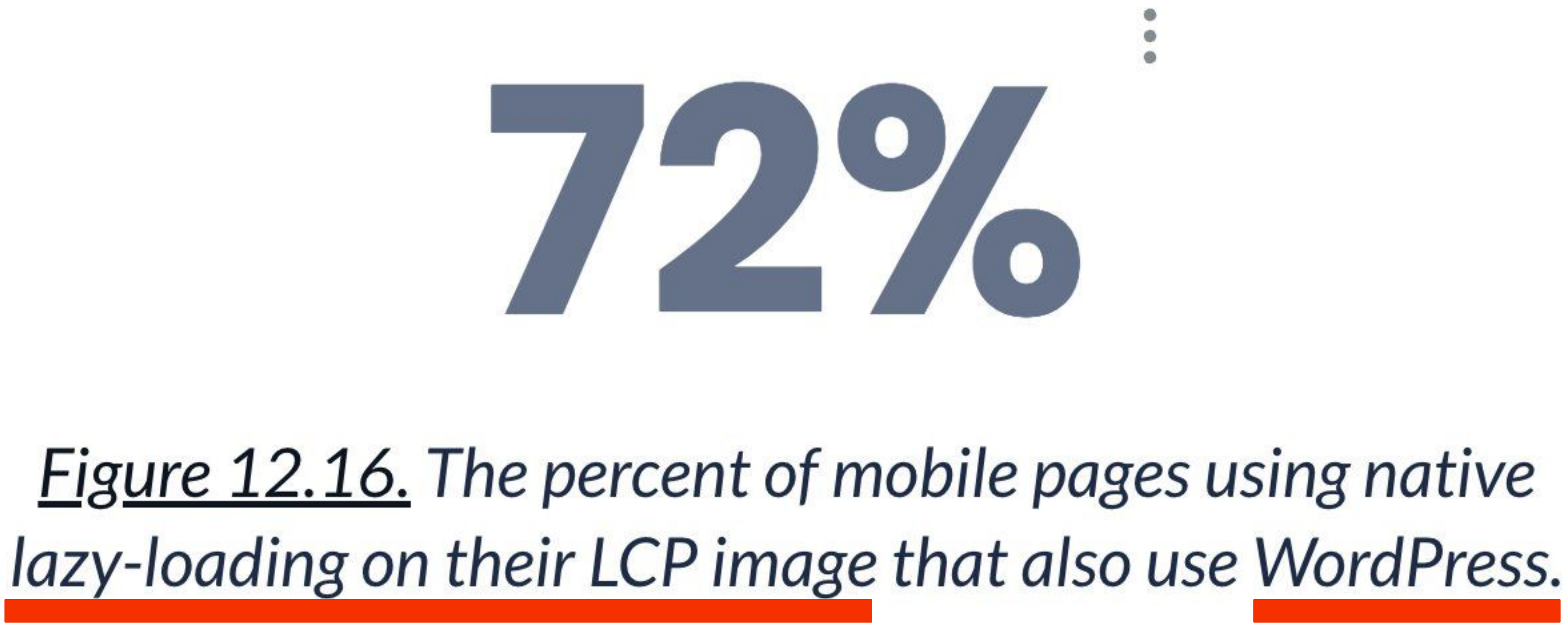
- /css/chunk-0112032d.cc09ddcf.css
- /css/chunk-0150f84c.6cbbfa5a.css
- /js/chunk-22accc54.21481e62.js
- /css/chunk-0222f9ab.69690fd5.css

Relevant Experiments

Remove Unu

This experimen
critical resourc

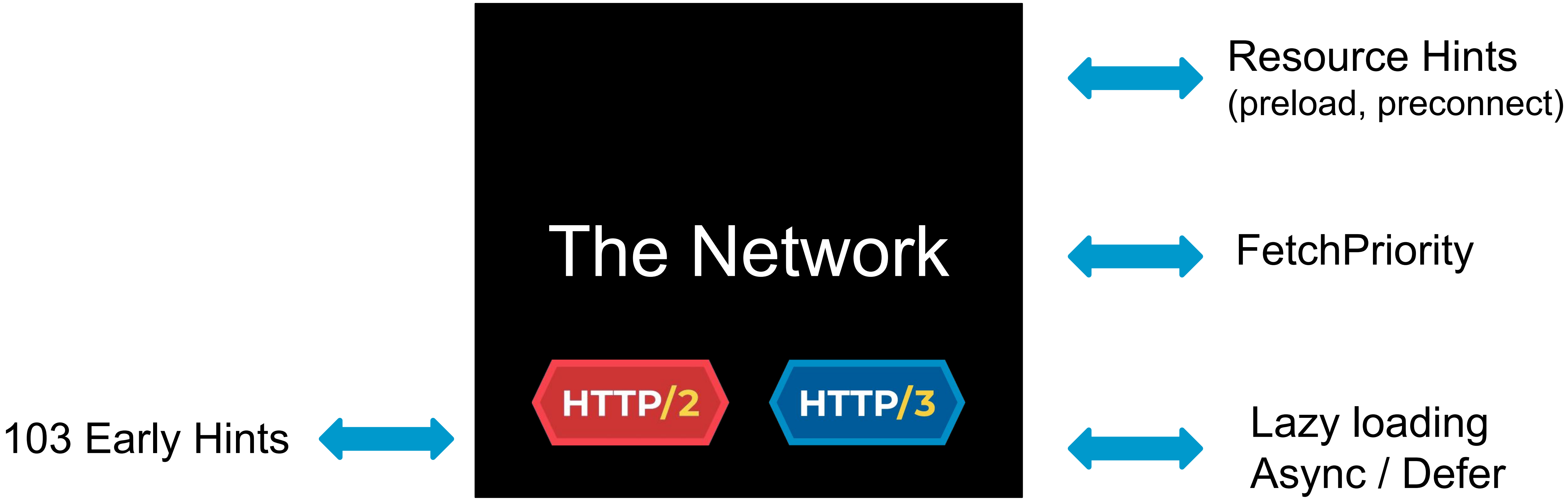
▶ **Assets inclu**



72%

Figure 12.16. The percent of mobile pages using native lazy-loading on their LCP image that also use WordPress.

https://web.dev/articles/lcp-lazy-loading
https://x.com/rick_viscomi/status/1585248419701874688?s=20
https://make.wordpress.org/core/2021/07/15/refining-wordpress-cores-lazy-loading-implementation/



The Network

HTTP/2

HTTP/3

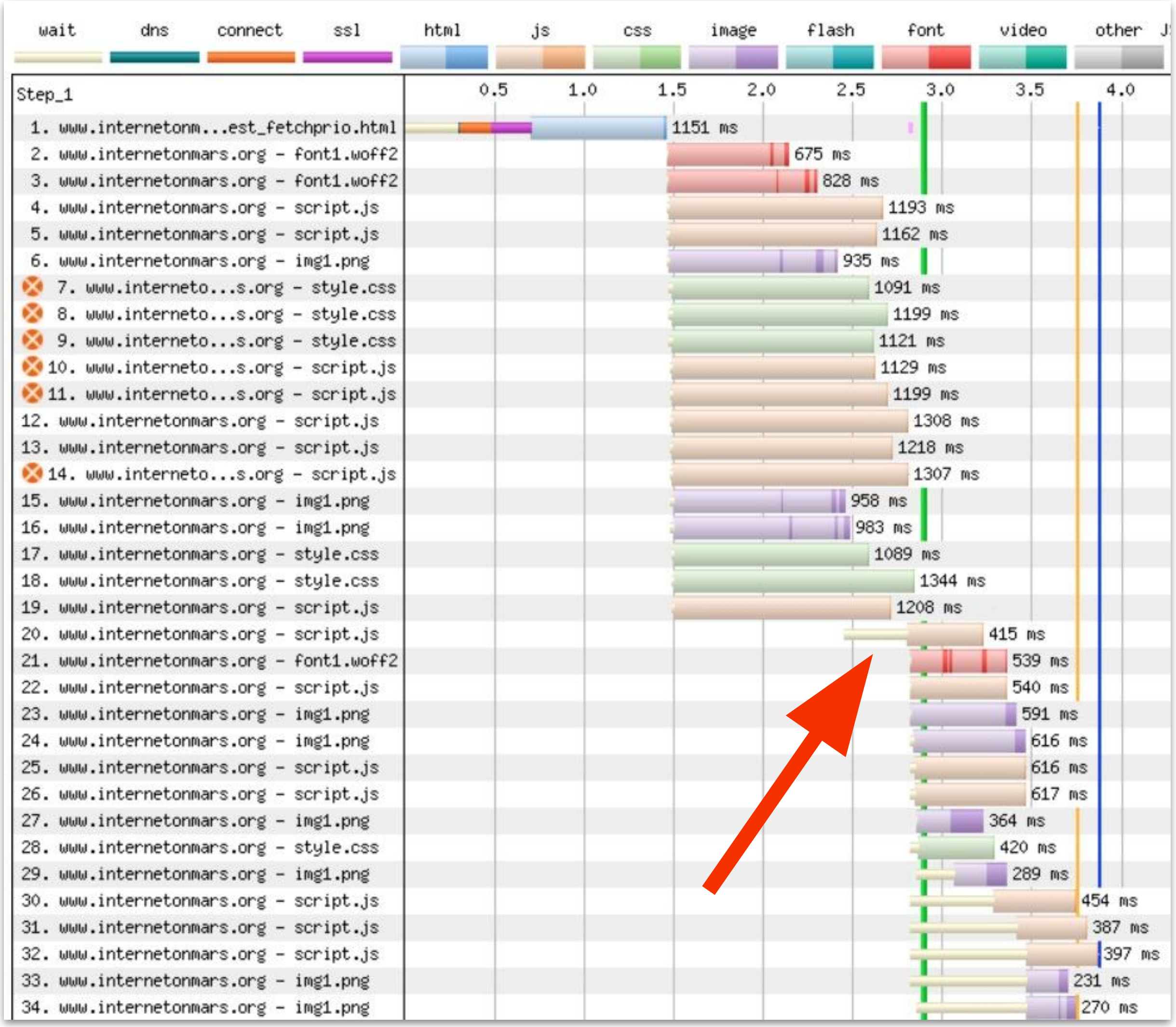
103 Early Hints

Resource Hints
(preload, preconnect)

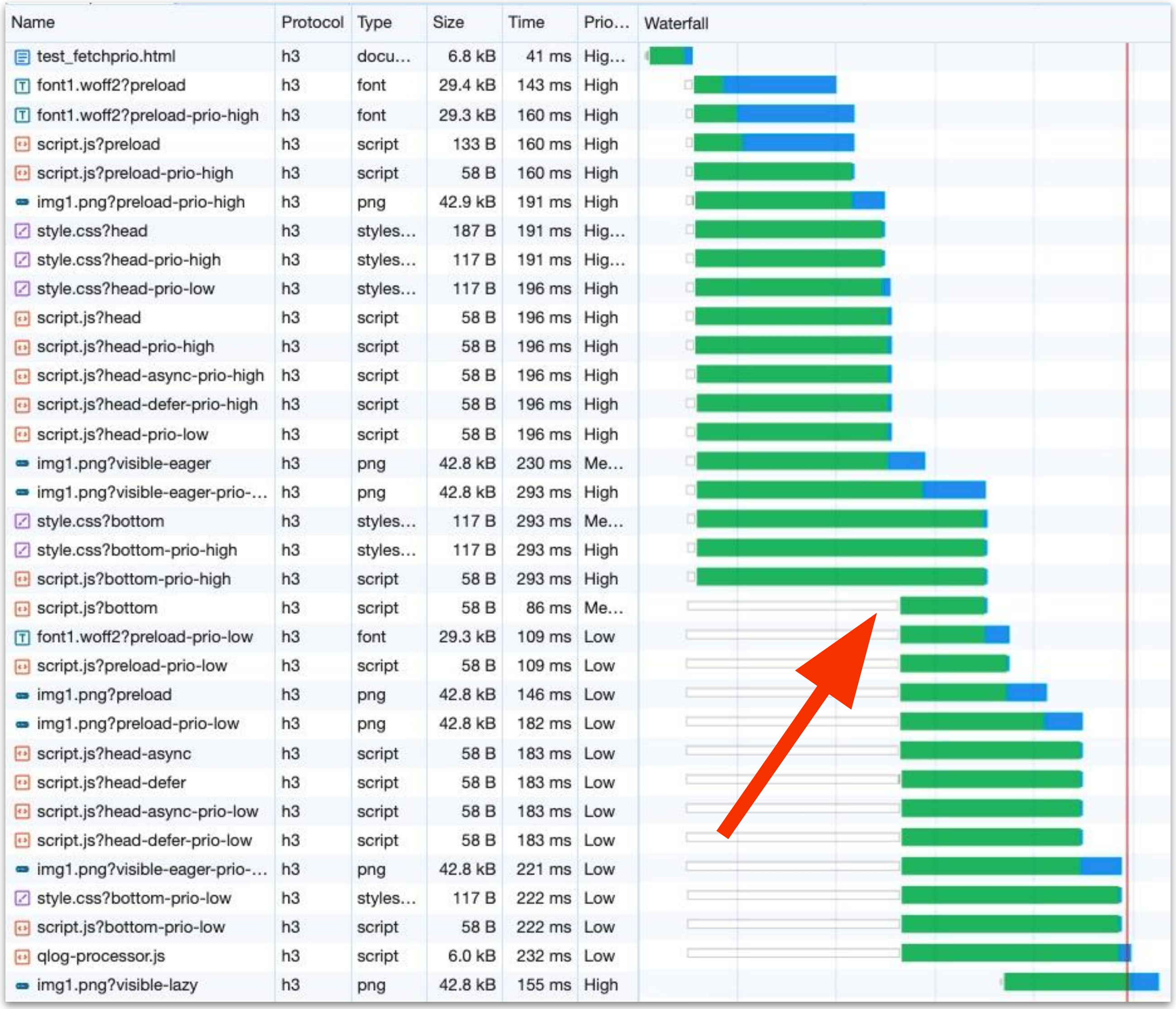
FetchPriority

Lazy loading
Async / Defer

Two-step waterfall



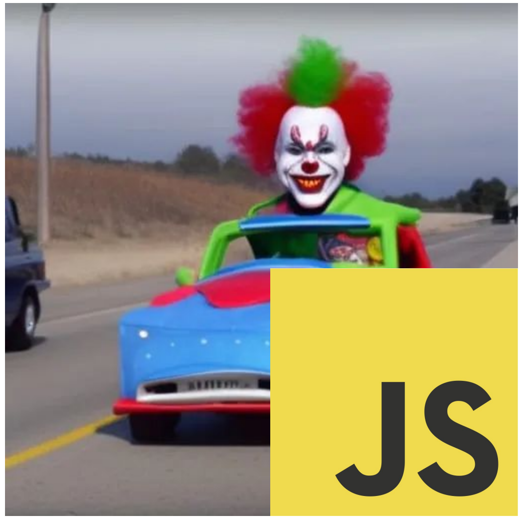
Step_1	
1. www.internetonm...est_fetchprio.html	1151 ms
2. www.internetonmars.org - font1.woff2	675 ms
3. www.internetonmars.org - font1.woff2	828 ms
4. www.internetonmars.org - script.js	1193 ms
5. www.internetonmars.org - script.js	1162 ms
6. www.internetonmars.org - img1.png	935 ms
⊗ 7. www.interneto...s.org - style.css	1091 ms
⊗ 8. www.interneto...s.org - style.css	1199 ms
⊗ 9. www.interneto...s.org - style.css	1121 ms
⊗ 10. www.interneto...s.org - script.js	1129 ms
⊗ 11. www.interneto...s.org - script.js	1199 ms
12. www.internetonmars.org - script.js	1308 ms
13. www.internetonmars.org - script.js	1218 ms
⊗ 14. www.interneto...s.org - script.js	1307 ms
15. www.internetonmars.org - img1.png	958 ms
16. www.internetonmars.org - img1.png	983 ms
17. www.internetonmars.org - style.css	1089 ms
18. www.internetonmars.org - style.css	1344 ms
19. www.internetonmars.org - script.js	1208 ms
20. www.internetonmars.org - script.js	415 ms
21. www.internetonmars.org - font1.woff2	539 ms
22. www.internetonmars.org - script.js	540 ms
23. www.internetonmars.org - img1.png	591 ms
24. www.internetonmars.org - img1.png	616 ms
25. www.internetonmars.org - script.js	616 ms
26. www.internetonmars.org - script.js	617 ms
27. www.internetonmars.org - img1.png	364 ms
28. www.internetonmars.org - style.css	420 ms
29. www.internetonmars.org - img1.png	289 ms
30. www.internetonmars.org - script.js	454 ms
31. www.internetonmars.org - script.js	387 ms
32. www.internetonmars.org - script.js	397 ms
33. www.internetonmars.org - img1.png	231 ms
34. www.internetonmars.org - img1.png	270 ms



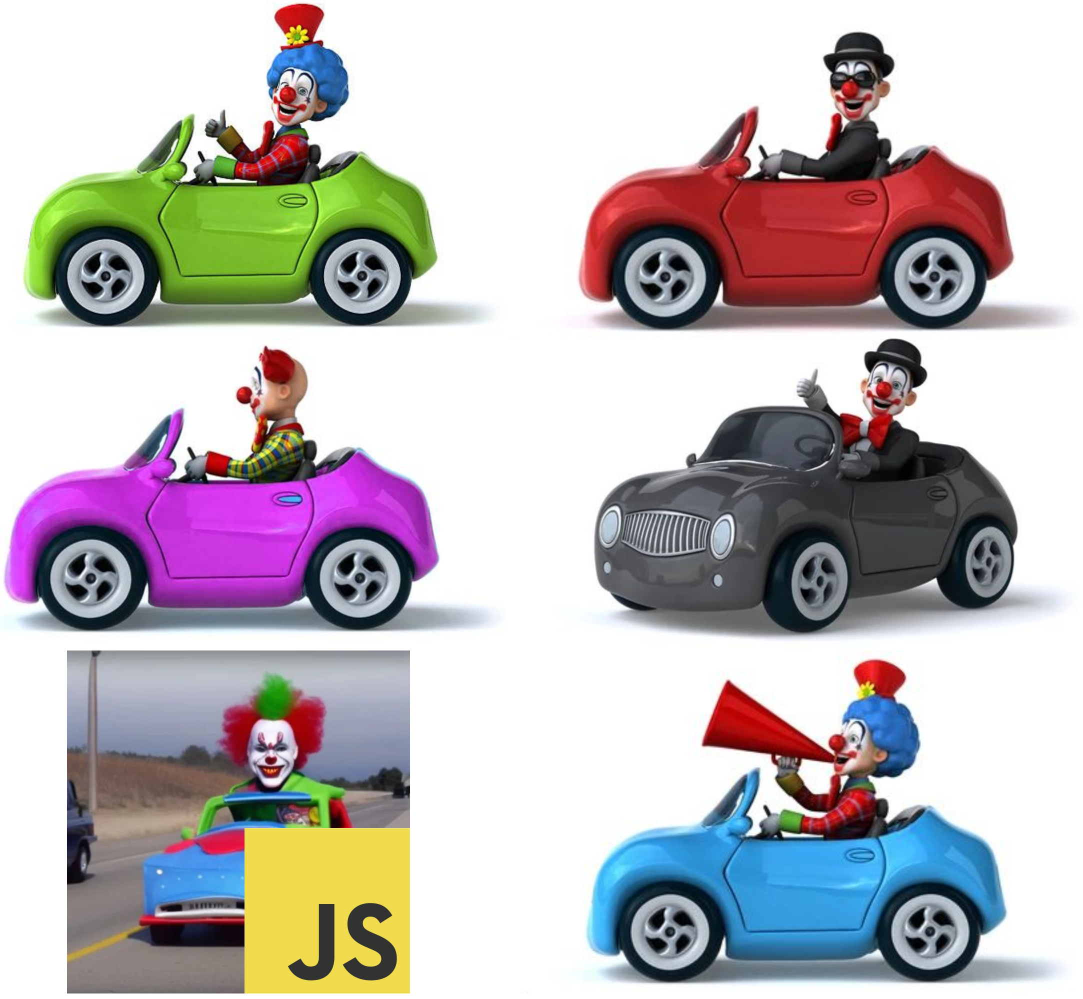
Name	Protocol	Type	Size	Time	Prio...	Waterfall
test_fetchprio.html	h3	docu...	6.8 kB	41 ms	Hig...	
font1.woff2?preload	h3	font	29.4 kB	143 ms	High	
font1.woff2?preload-prio-high	h3	font	29.3 kB	160 ms	High	
script.js?preload	h3	script	133 B	160 ms	High	
script.js?preload-prio-high	h3	script	58 B	160 ms	High	
img1.png?preload-prio-high	h3	png	42.9 kB	191 ms	High	
style.css?head	h3	styles...	187 B	191 ms	Hig...	
style.css?head-prio-high	h3	styles...	117 B	191 ms	Hig...	
style.css?head-prio-low	h3	styles...	117 B	196 ms	High	
script.js?head	h3	script	58 B	196 ms	High	
script.js?head-prio-high	h3	script	58 B	196 ms	High	
script.js?head-async-prio-high	h3	script	58 B	196 ms	High	
script.js?head-defer-prio-high	h3	script	58 B	196 ms	High	
script.js?head-prio-low	h3	script	58 B	196 ms	High	
img1.png?visible-eager	h3	png	42.8 kB	230 ms	Me...	
img1.png?visible-eager-prio-...	h3	png	42.8 kB	293 ms	High	
style.css?bottom	h3	styles...	117 B	293 ms	Me...	
style.css?bottom-prio-high	h3	styles...	117 B	293 ms	High	
script.js?bottom-prio-high	h3	script	58 B	293 ms	High	
script.js?bottom	h3	script	58 B	86 ms	Me...	
font1.woff2?preload-prio-low	h3	font	29.3 kB	109 ms	Low	
script.js?preload-prio-low	h3	script	58 B	109 ms	Low	
img1.png?preload	h3	png	42.8 kB	146 ms	Low	
img1.png?preload-prio-low	h3	png	42.8 kB	182 ms	Low	
script.js?head-async	h3	script	58 B	183 ms	Low	
script.js?head-defer	h3	script	58 B	183 ms	Low	
script.js?head-async-prio-low	h3	script	58 B	183 ms	Low	
script.js?head-defer-prio-low	h3	script	58 B	183 ms	Low	
img1.png?visible-eager-prio-...	h3	png	42.8 kB	221 ms	Low	
style.css?bottom-prio-low	h3	styles...	117 B	222 ms	Low	
script.js?bottom-prio-low	h3	script	58 B	222 ms	Low	
qlog-processor.js	h3	script	6.0 kB	232 ms	Low	
img1.png?visible-lazy	h3	png	42.8 kB	155 ms	High	

HTTP/1.1 is limited to 1 resource per connection

HTTP/1.1 is limited to 1 resource per connection



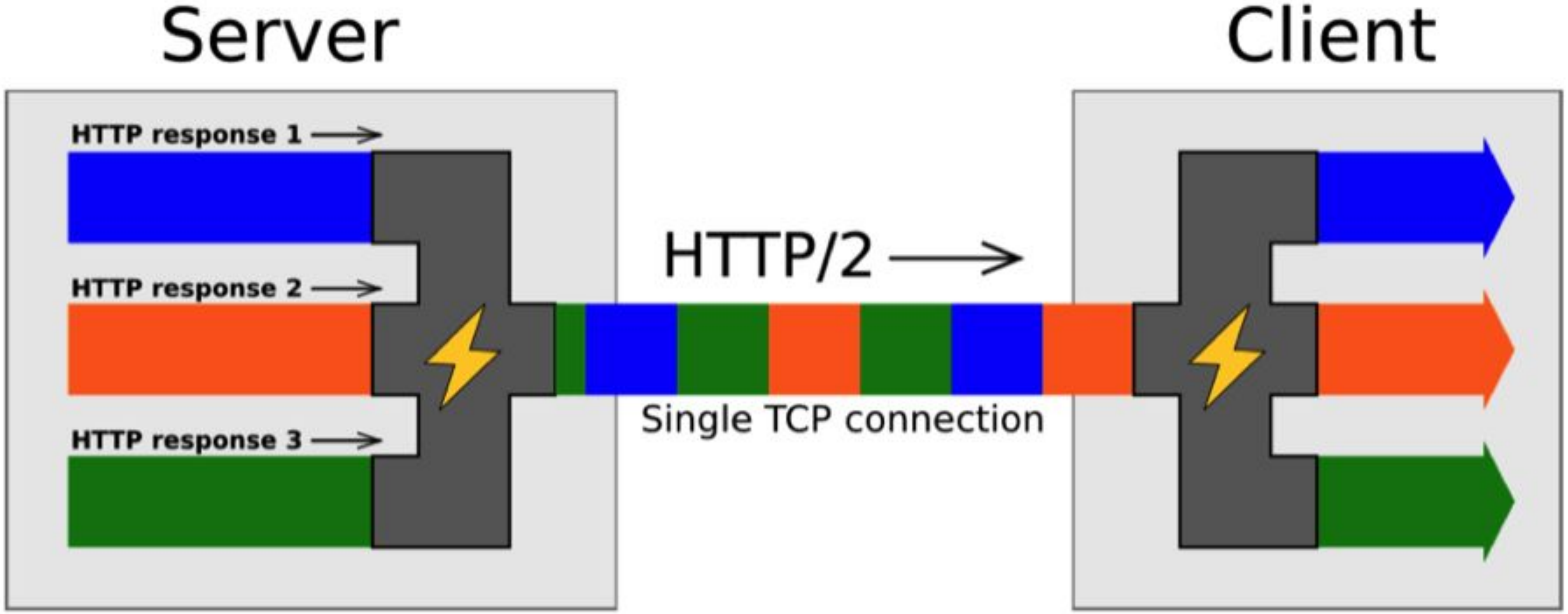
A ClownCar named Desire

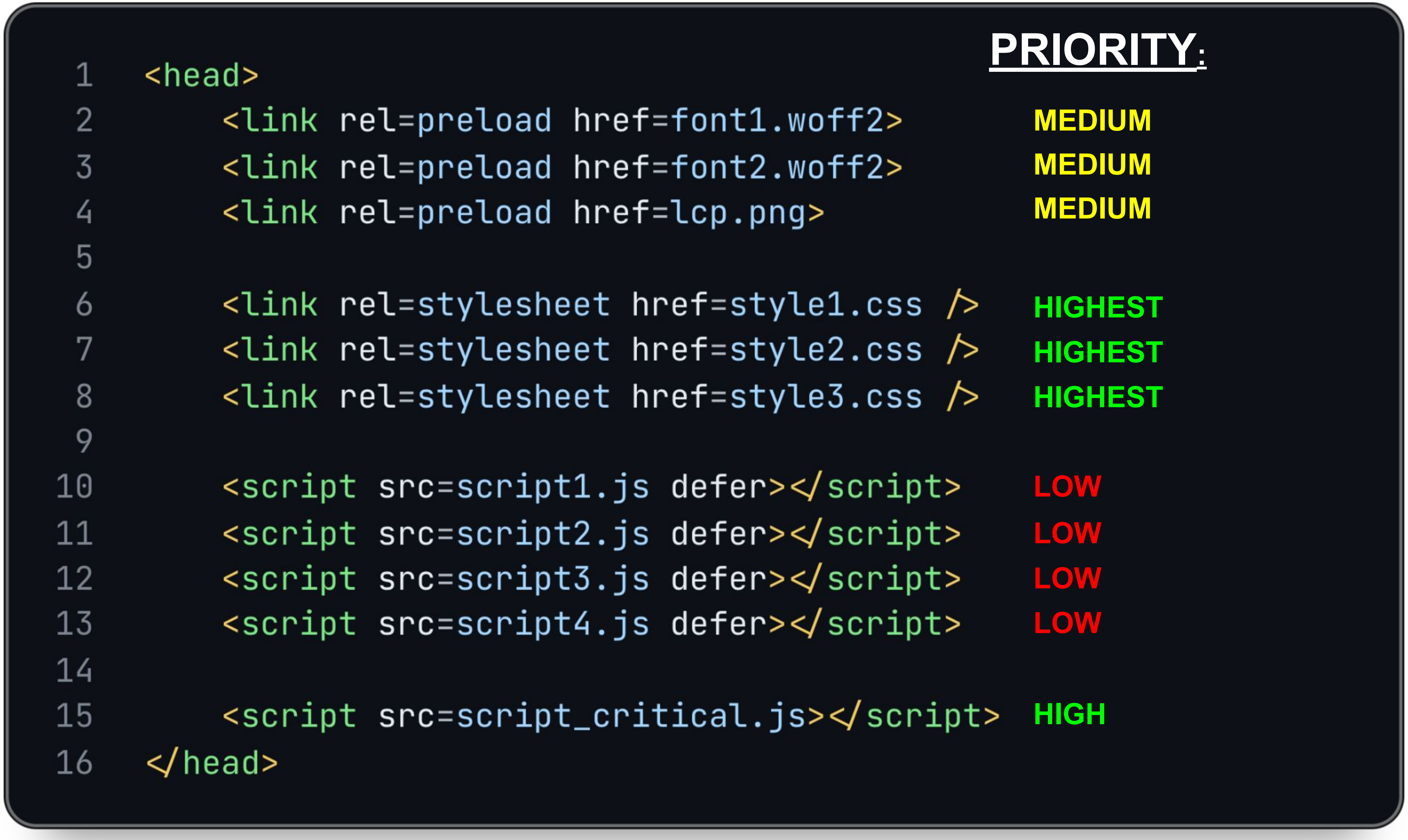


HTTP/1.1

HTTP/2 and /3

HTTP/2 and /3 Multiplexing

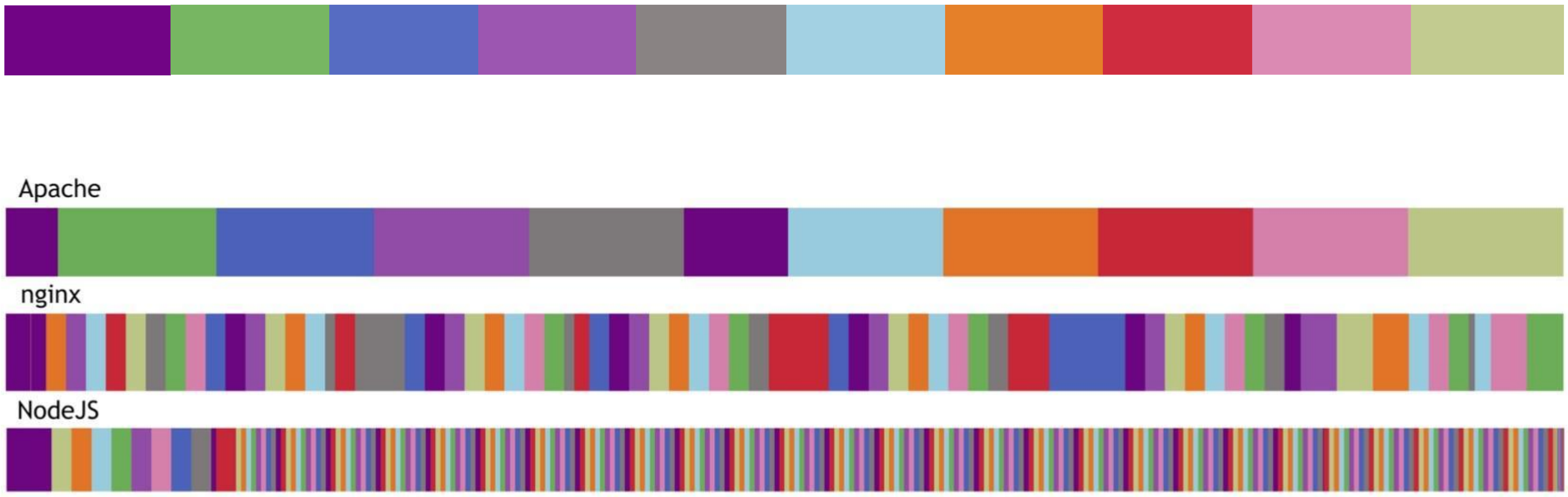




```
                                                        PRIORITY:
<head>
    <link rel=preload href=font1.woff2>                 MEDIUM
    <link rel=preload href=font2.woff2>                 MEDIUM
    <link rel=preload href=lcp.png>                     MEDIUM

    <link rel=stylesheet href=style1.css />             HIGHEST
    <link rel=stylesheet href=style2.css />             HIGHEST
    <link rel=stylesheet href=style3.css />             HIGHEST

    <script src=script1.js defer></script>              LOW
    <script src=script2.js defer></script>              LOW
    <script src=script3.js defer></script>              LOW
    <script src=script4.js defer></script>              LOW

    <script src=script_critical.js></script>            HIGH
</head>
```

(HTTP/2) Servers often don’t listen to browsers…

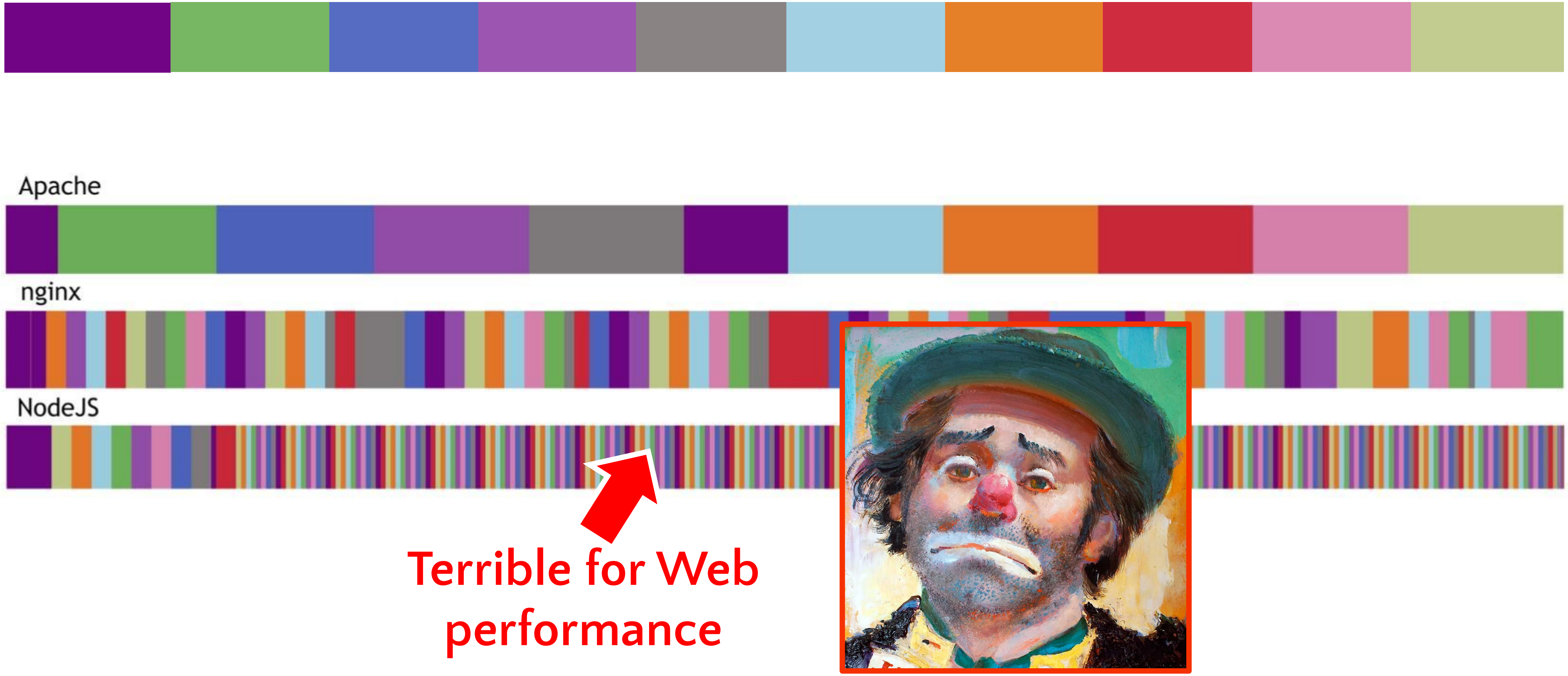
Browser instructions:



https://github.com/andydavies/http2-prioritization-issues
https://www.researchgate.net/publication/347519865_Debugging_Modern_Web_Protocols

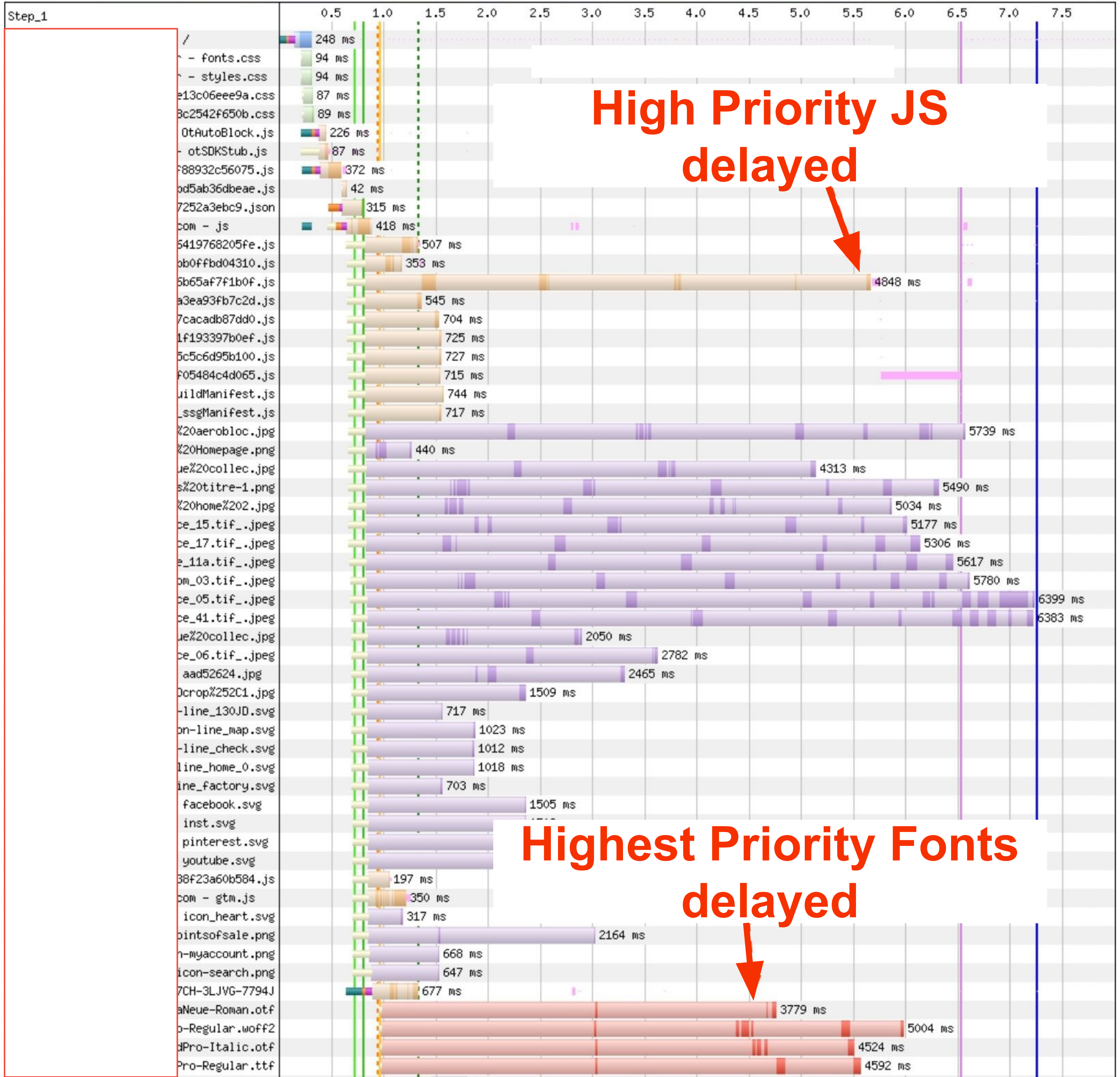
(HTTP/2) Servers often don’t listen to browsers…

Browser instructions:



Terrible for Web performance

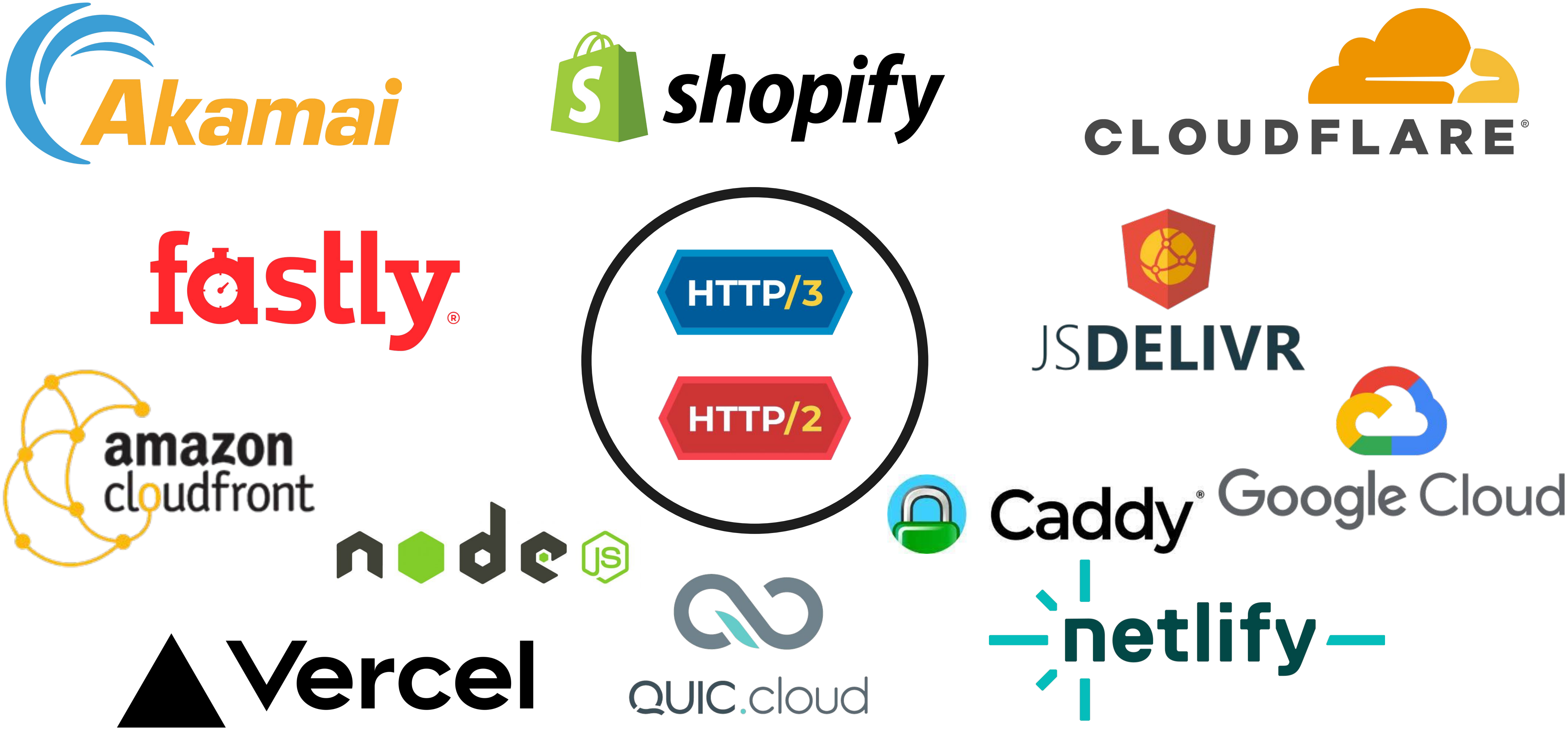
https://github.com/andydavies/http2-prioritization-issues
https://www.researchgate.net/publication/347519865_Debugging_Modern_Web_Protocols



High Priority JS delayed

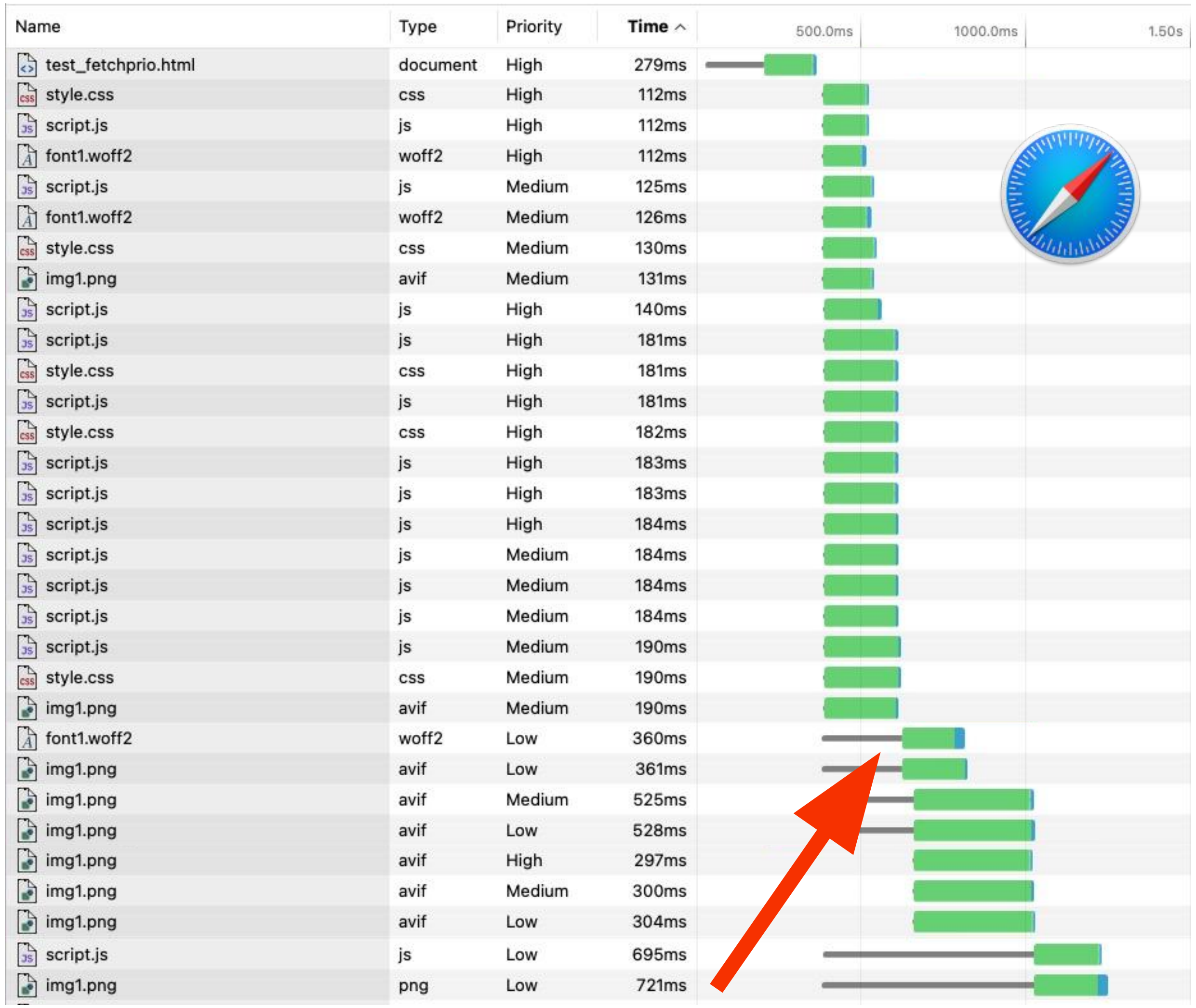
Highest Priority Fonts delayed

Only 2 of these companies do it (100%) correctly…

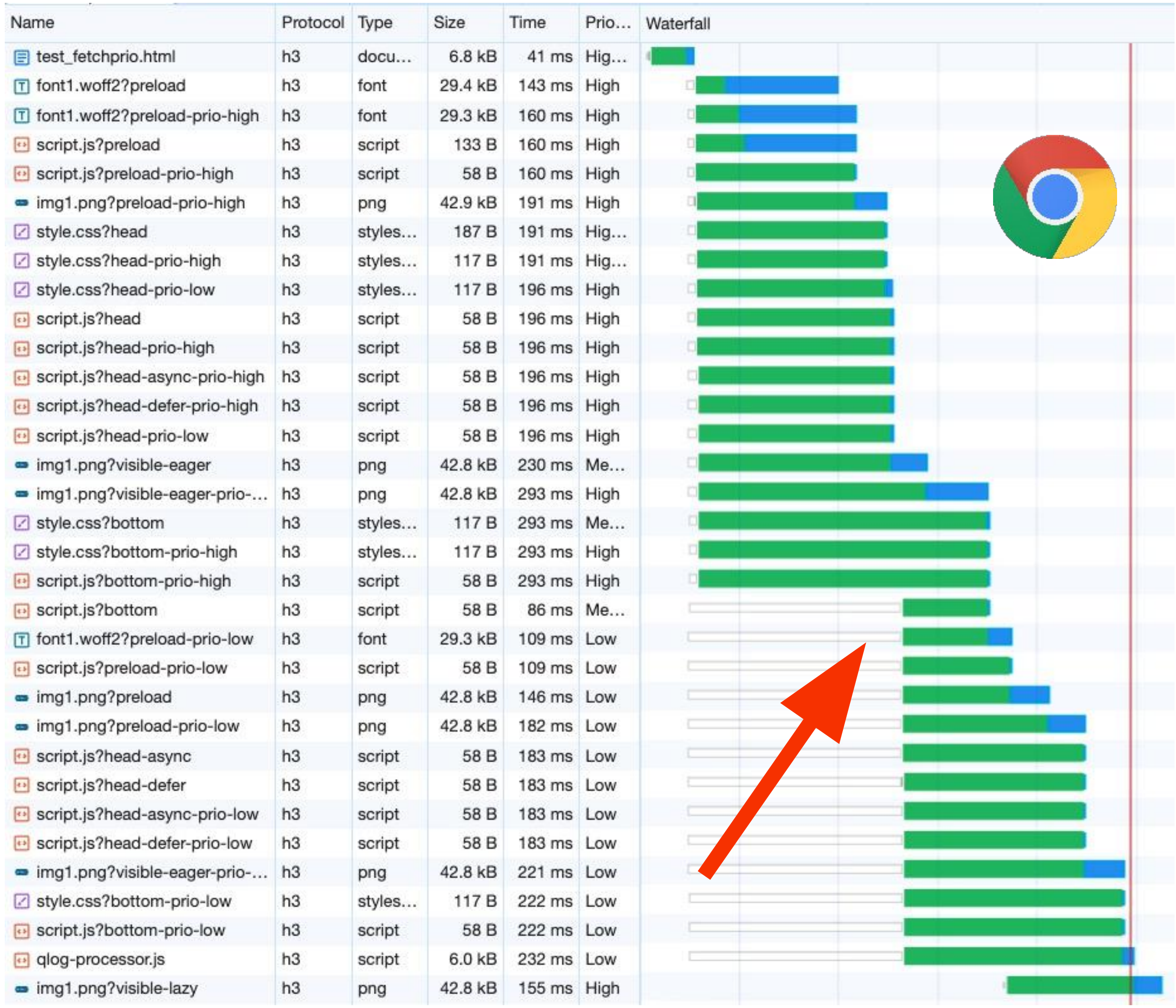


https://jherbots.info/public_media/research/anrw2024_h3-eps-in-the-wild_authorversion.pdf

Two-step waterfall even with HTTP/2 and HTTP/3!

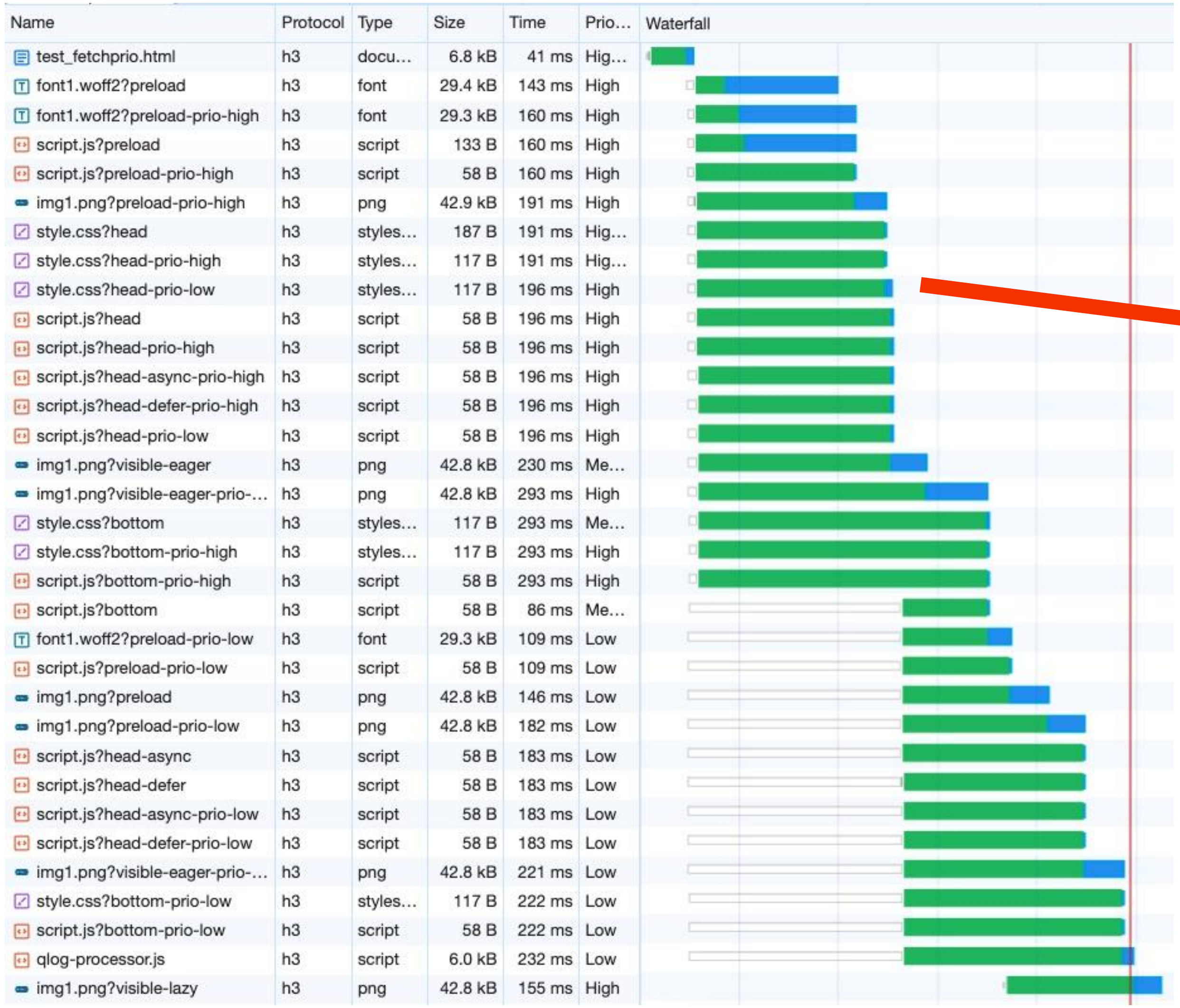


Name	Type	Priority	Time ^
test_fetchprio.html	document	High	279ms
style.css	css	High	112ms
script.js	js	High	112ms
font1.woff2	woff2	High	112ms
script.js	js	Medium	125ms
font1.woff2	woff2	Medium	126ms
style.css	css	Medium	130ms
img1.png	avif	Medium	131ms
script.js	js	High	140ms
script.js	js	High	181ms
style.css	css	High	181ms
script.js	js	High	181ms
style.css	css	High	182ms
script.js	js	High	183ms
script.js	js	High	183ms
script.js	js	High	184ms
script.js	js	Medium	184ms
script.js	js	Medium	184ms
script.js	js	Medium	184ms
script.js	js	Medium	190ms
style.css	css	Medium	190ms
img1.png	avif	Medium	190ms
font1.woff2	woff2	Low	360ms
img1.png	avif	Low	361ms
img1.png	avif	Medium	525ms
img1.png	avif	Low	528ms
img1.png	avif	High	297ms
img1.png	avif	Medium	300ms
img1.png	avif	Low	304ms
script.js	js	Low	695ms
img1.png	png	Low	721ms



Name	Protocol	Type	Size	Time	Prio...	Waterfall
test_fetchprio.html	h3	docu...	6.8 kB	41 ms	Hig...	
font1.woff2?preload	h3	font	29.4 kB	143 ms	High	
font1.woff2?preload-prio-high	h3	font	29.3 kB	160 ms	High	
script.js?preload	h3	script	133 B	160 ms	High	
script.js?preload-prio-high	h3	script	58 B	160 ms	High	
img1.png?preload-prio-high	h3	png	42.9 kB	191 ms	High	
style.css?head	h3	styles...	187 B	191 ms	Hig...	
style.css?head-prio-high	h3	styles...	117 B	191 ms	Hig...	
style.css?head-prio-low	h3	styles...	117 B	196 ms	High	
script.js?head	h3	script	58 B	196 ms	High	
script.js?head-prio-high	h3	script	58 B	196 ms	High	
script.js?head-async-prio-high	h3	script	58 B	196 ms	High	
script.js?head-defer-prio-high	h3	script	58 B	196 ms	High	
script.js?head-prio-low	h3	script	58 B	196 ms	High	
img1.png?visible-eager	h3	png	42.8 kB	230 ms	Me...	
img1.png?visible-eager-prio-...	h3	png	42.8 kB	293 ms	High	
style.css?bottom	h3	styles...	117 B	293 ms	Me...	
style.css?bottom-prio-high	h3	styles...	117 B	293 ms	High	
script.js?bottom-prio-high	h3	script	58 B	293 ms	High	
script.js?bottom	h3	script	58 B	86 ms	Me...	
font1.woff2?preload-prio-low	h3	font	29.3 kB	109 ms	Low	
script.js?preload-prio-low	h3	script	58 B	109 ms	Low	
img1.png?preload	h3	png	42.8 kB	146 ms	Low	
img1.png?preload-prio-low	h3	png	42.8 kB	182 ms	Low	
script.js?head-async	h3	script	58 B	183 ms	Low	
script.js?head-defer	h3	script	58 B	183 ms	Low	
script.js?head-async-prio-low	h3	script	58 B	183 ms	Low	
script.js?head-defer-prio-low	h3	script	58 B	183 ms	Low	
img1.png?visible-eager-prio-...	h3	png	42.8 kB	221 ms	Low	
style.css?bottom-prio-low	h3	styles...	117 B	222 ms	Low	
script.js?bottom-prio-low	h3	script	58 B	222 ms	Low	
qlog-processor.js	h3	script	6.0 kB	232 ms	Low	
img1.png?visible-lazy	h3	png	42.8 kB	155 ms	High	

What if the BROWSER gets it wrong…



Name	Protocol	Type	Size	Time	Prio…	Waterfall
test_fetchprio.html	h3	docu…	6.8 kB	41 ms	Hig…	
font1.woff2?preload	h3	font	29.4 kB	143 ms	High	
font1.woff2?preload-prio-high	h3	font	29.3 kB	160 ms	High	
script.js?preload	h3	script	133 B	160 ms	High	
script.js?preload-prio-high	h3	script	58 B	160 ms	High	
img1.png?preload-prio-high	h3	png	42.9 kB	191 ms	High	
style.css?head	h3	styles…	187 B	191 ms	Hig…	
style.css?head-prio-high	h3	styles…	117 B	191 ms	Hig…	
style.css?head-prio-low	h3	styles…	117 B	196 ms	High	
script.js?head	h3	script	58 B	196 ms	High	
script.js?head-prio-high	h3	script	58 B	196 ms	High	
script.js?head-async-prio-high	h3	script	58 B	196 ms	High	
script.js?head-defer-prio-high	h3	script	58 B	196 ms	High	
script.js?head-prio-low	h3	script	58 B	196 ms	High	
img1.png?visible-eager	h3	png	42.8 kB	230 ms	Me…	
img1.png?visible-eager-prio-…	h3	png	42.8 kB	293 ms	High	
style.css?bottom	h3	styles…	117 B	293 ms	Me…	
style.css?bottom-prio-high	h3	styles…	117 B	293 ms	High	
script.js?bottom-prio-high	h3	script	58 B	293 ms	High	
script.js?bottom	h3	script	58 B	86 ms	Me…	
font1.woff2?preload-prio-low	h3	font	29.3 kB	109 ms	Low	
script.js?preload-prio-low	h3	script	58 B	109 ms	Low	
img1.png?preload	h3	png	42.8 kB	146 ms	Low	
img1.png?preload-prio-low	h3	png	42.8 kB	182 ms	Low	
script.js?head-async	h3	script	58 B	183 ms	Low	
script.js?head-defer	h3	script	58 B	183 ms	Low	
script.js?head-async-prio-low	h3	script	58 B	183 ms	Low	
script.js?head-defer-prio-low	h3	script	58 B	183 ms	Low	
img1.png?visible-eager-prio-…	h3	png	42.8 kB	221 ms	Low	
style.css?bottom-prio-low	h3	styles…	117 B	222 ms	Low	
script.js?bottom-prio-low	h3	script	58 B	222 ms	Low	
qlog-processor.js	h3	script	6.0 kB	232 ms	Low	
img1.png?visible-lazy	h3	png	42.8 kB	155 ms	High	

This should actually be down there

(or vice versa)

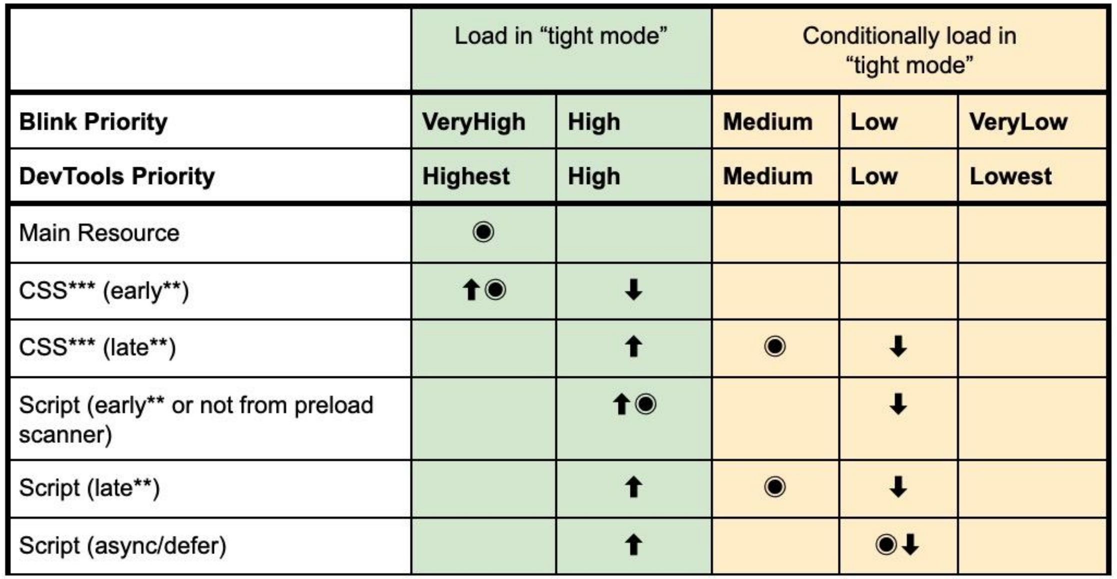
Resource Fetch Prioritization and Scheduling in Chrome

Author: Patrick Meenan

August 5, 2015 (Updated June 27, 2022)

Current State

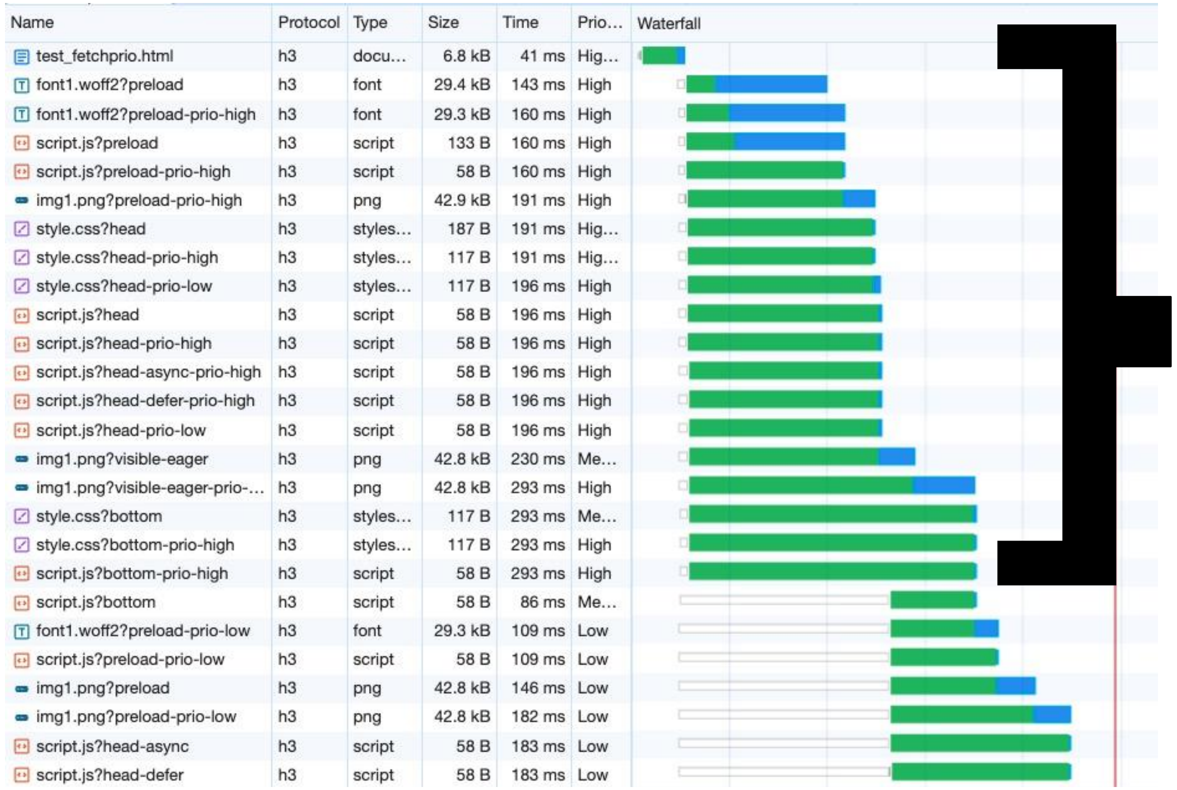
As of April 2022, the table below represents how all resources in Chrome are handled:



	Load in “tight mode”		Conditionally load in “tight mode”		
Blink Priority	**VeryHigh**	**High**	**Medium**	**Low**	**VeryLow**
DevTools Priority	**Highest**	**High**	**Medium**	**Low**	**Lowest**
Main Resource	◉				
CSS*** (early**)	⬆◉	⬇			
CSS*** (late**)		⬆	◉	⬇	
Script (early** or not from preload scanner)		⬆◉		⬇	
Script (late**)		⬆	◉	⬇	
Script (async/defer)		⬆		◉⬇	

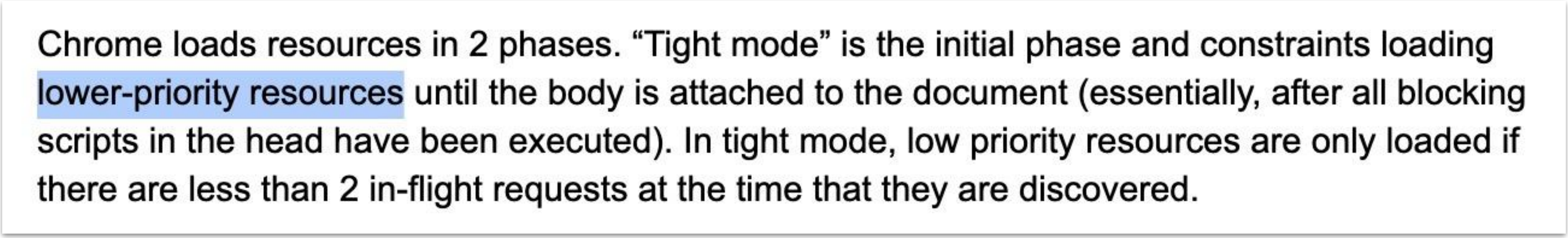
https://web.dev/articles/fetch-priority
https://imkev.dev/fetchpriority-opportunity
https://docs.google.com/document/d/1bCDuq9H1ih9iNjgzyAL0gpwNFiEP4TZS-YLRp_RuMIc

MIKE LARSON
2019



Name	Protocol	Type	Size	Time	Prio…	Waterfall
test_fetchprio.html	h3	docu…	6.8 kB	41 ms	Hig…	
font1.woff2?preload	h3	font	29.4 kB	143 ms	High	
font1.woff2?preload-prio-high	h3	font	29.3 kB	160 ms	High	
script.js?preload	h3	script	133 B	160 ms	High	
script.js?preload-prio-high	h3	script	58 B	160 ms	High	
img1.png?preload-prio-high	h3	png	42.9 kB	191 ms	High	
style.css?head	h3	styles…	187 B	191 ms	Hig…	
style.css?head-prio-high	h3	styles…	117 B	191 ms	Hig…	
style.css?head-prio-low	h3	styles…	117 B	196 ms	High	
script.js?head	h3	script	58 B	196 ms	High	
script.js?head-prio-high	h3	script	58 B	196 ms	High	
script.js?head-async-prio-high	h3	script	58 B	196 ms	High	
script.js?head-defer-prio-high	h3	script	58 B	196 ms	High	
script.js?head-prio-low	h3	script	58 B	196 ms	High	
img1.png?visible-eager	h3	png	42.8 kB	230 ms	Me…	
img1.png?visible-eager-prio-…	h3	png	42.8 kB	293 ms	High	
style.css?bottom	h3	styles…	117 B	293 ms	Me…	
style.css?bottom-prio-high	h3	styles…	117 B	293 ms	High	
script.js?bottom-prio-high	h3	script	58 B	293 ms	High	
script.js?bottom	h3	script	58 B	86 ms	Me…	
font1.woff2?preload-prio-low	h3	font	29.3 kB	109 ms	Low	
script.js?preload-prio-low	h3	script	58 B	109 ms	Low	
img1.png?preload	h3	png	42.8 kB	146 ms	Low	
img1.png?preload-prio-low	h3	png	42.8 kB	182 ms	Low	
script.js?head-async	h3	script	58 B	183 ms	Low	
script.js?head-defer	h3	script	58 B	183 ms	Low	

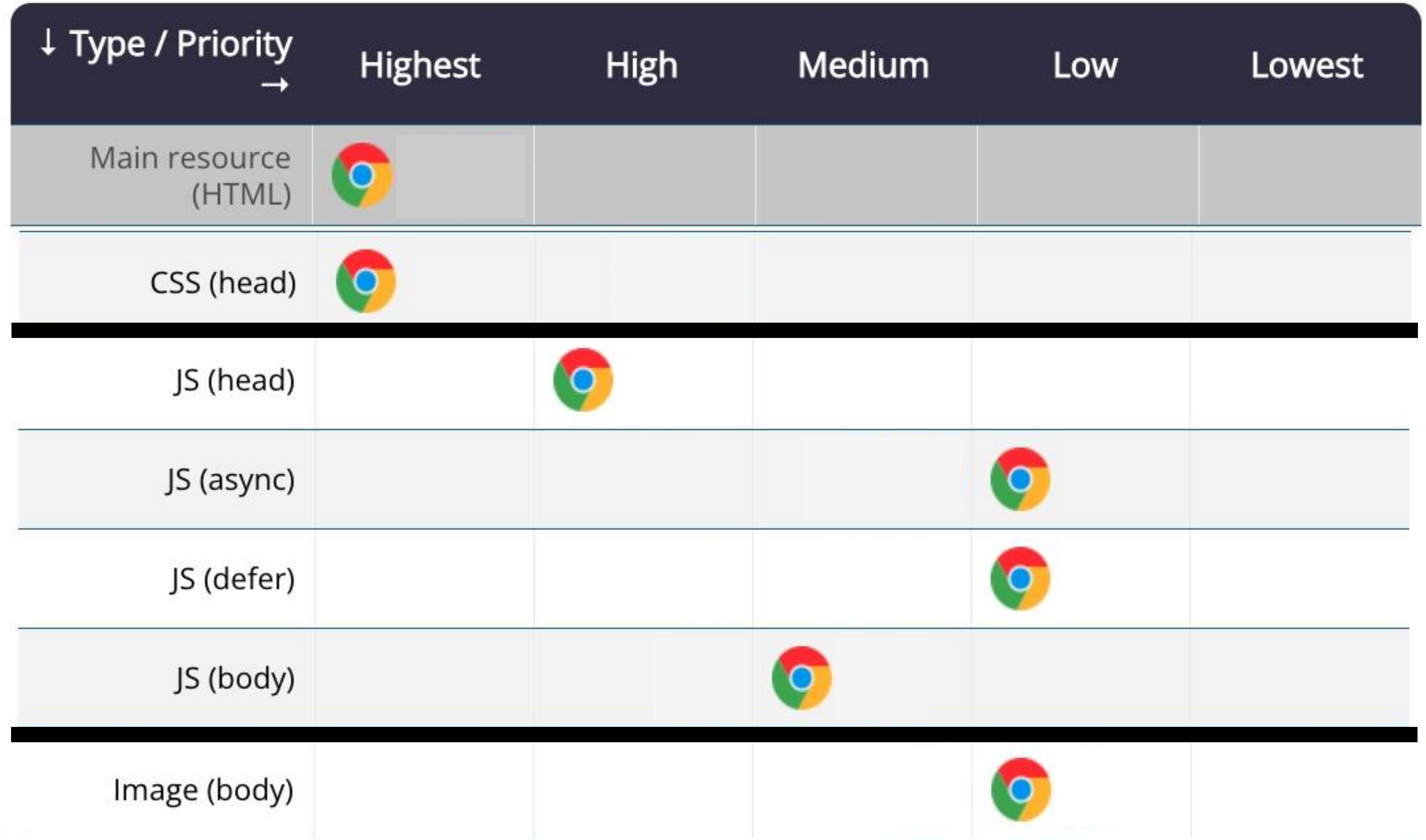
Tight Mode



Chrome loads resources in 2 phases. “Tight mode” is the initial phase and constraints loading lower-priority resources until the body is attached to the document (essentially, after all blocking scripts in the head have been executed). In tight mode, low priority resources are only loaded if there are less than 2 in-flight requests at the time that they are discovered.

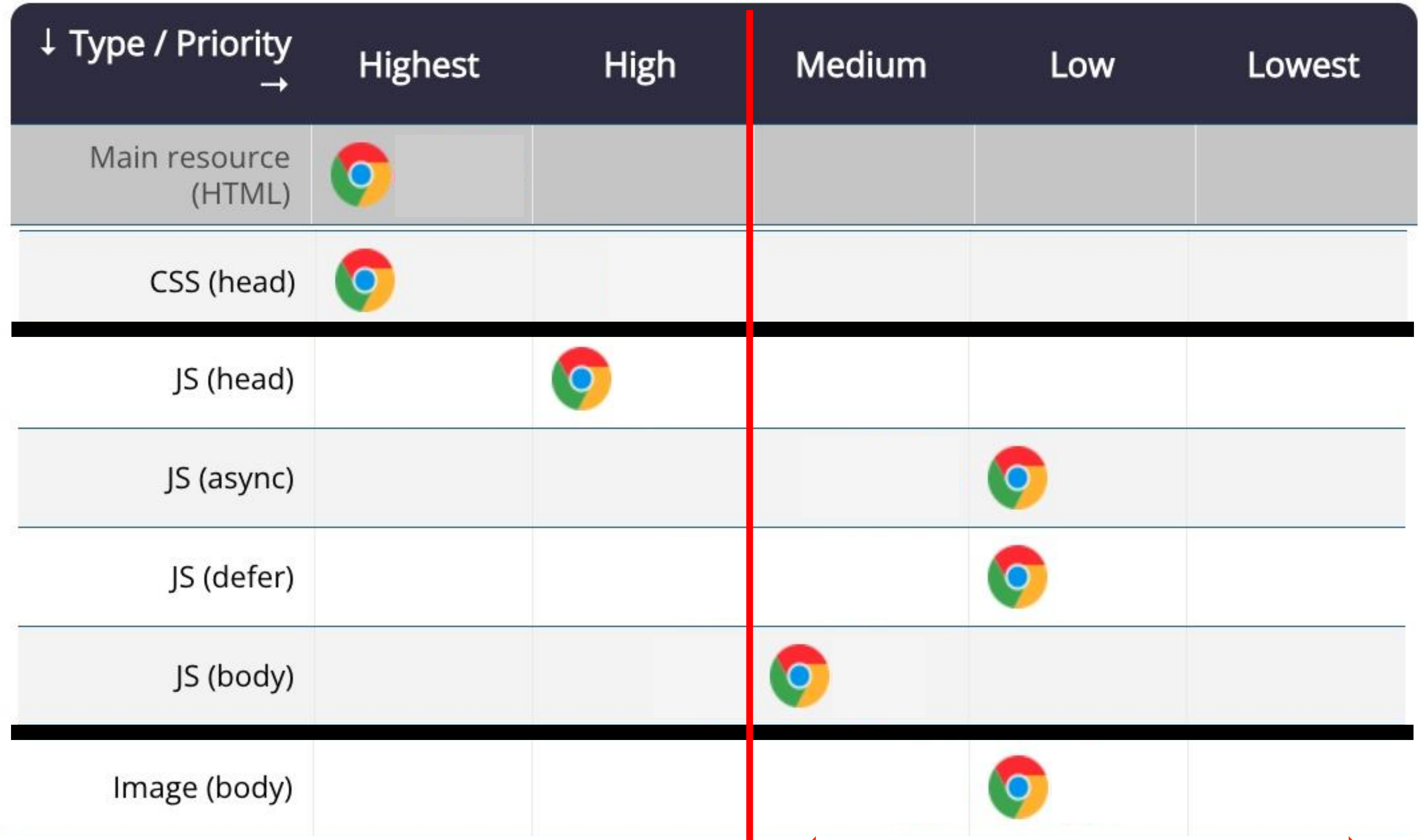
https://docs.google.com/document/d/1bCDuq9H1ih9iNjgzyAL0gpwNFiEP4TZS-YLRp_RuMlc

Priority: *where* stuff is in HTML and *how* it's loaded



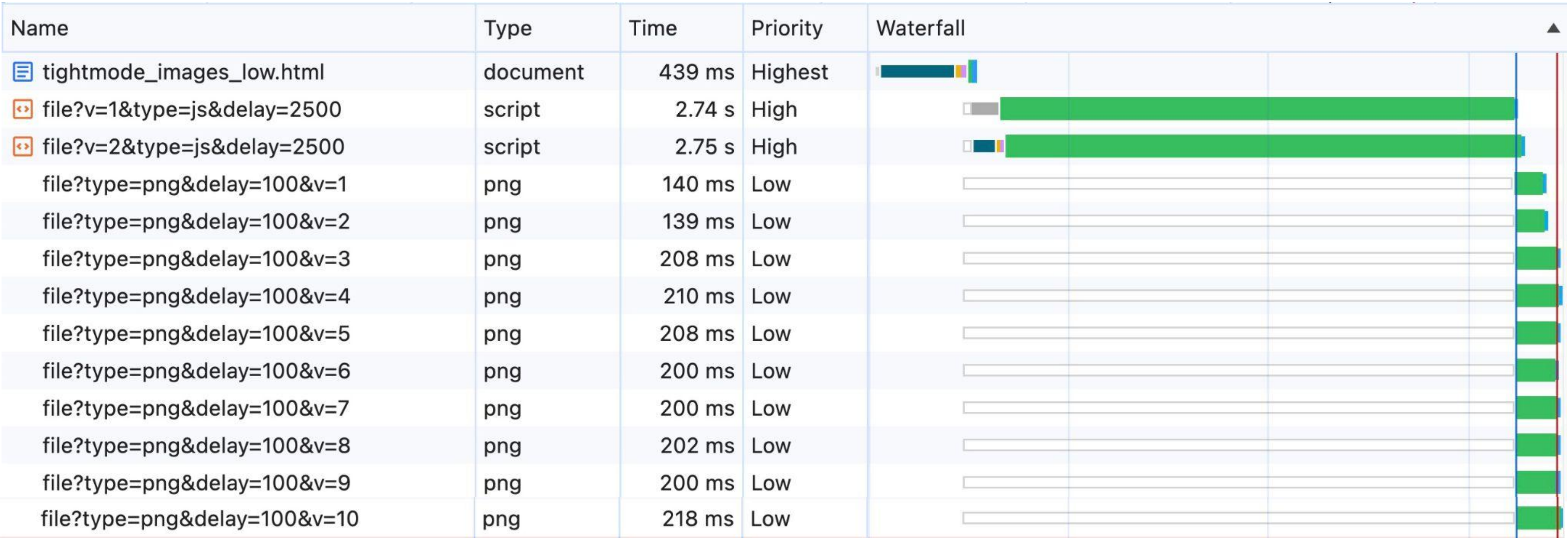
↓ Type / Priority →	Highest	High	Medium	Low	Lowest
Main resource (HTML)	●				
CSS (head)	●				
JS (head)		●			
JS (async)				●	
JS (defer)				●	
JS (body)			●		
Image (body)				●	

“Lower-priority”: medium + low + lowest



↓ Type / Priority →	Highest	High	Medium	Low	Lowest
Main resource (HTML)	●				
CSS (head)	●				
JS (head)		●			
JS (async)				●	
JS (defer)				●	
JS (body)			●		
Image (body)				●	

Actively delayed

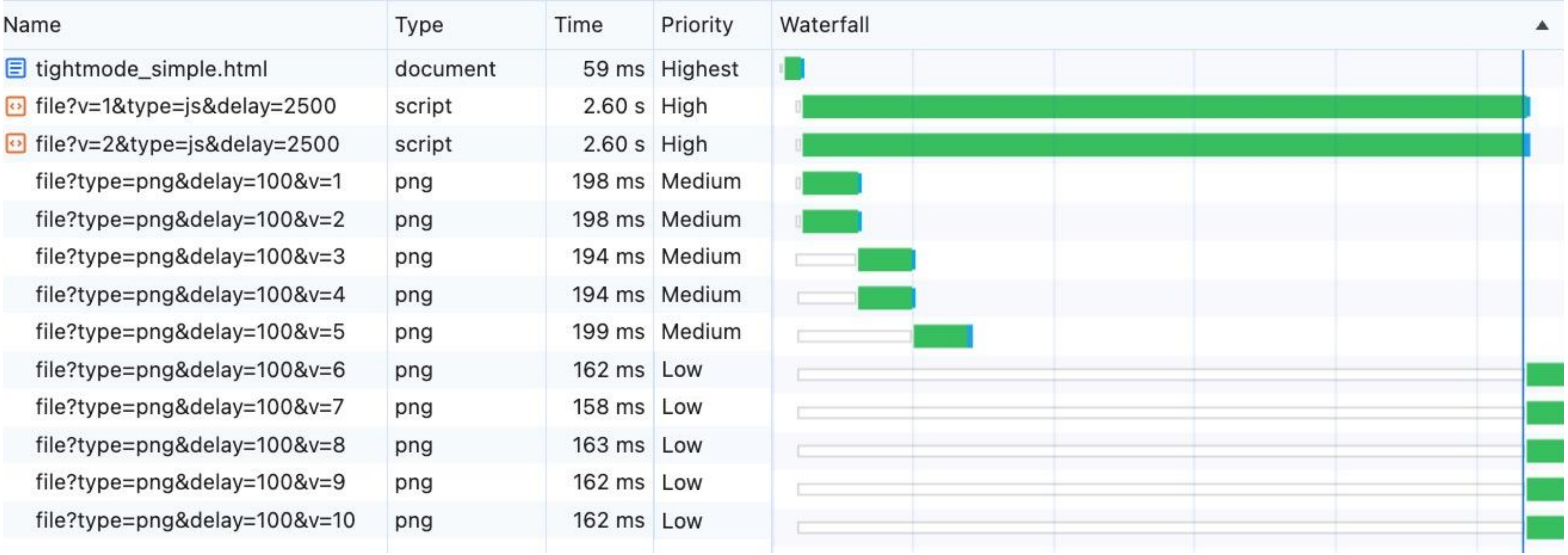


Name	Type	Time	Priority	Waterfall
tightmode_images_low.html	document	439 ms	Highest	
file?v=1&type=js&delay=2500	script	2.74 s	High	
file?v=2&type=js&delay=2500	script	2.75 s	High	
file?type=png&delay=100&v=1	png	140 ms	Low	
file?type=png&delay=100&v=2	png	139 ms	Low	
file?type=png&delay=100&v=3	png	208 ms	Low	
file?type=png&delay=100&v=4	png	210 ms	Low	
file?type=png&delay=100&v=5	png	208 ms	Low	
file?type=png&delay=100&v=6	png	200 ms	Low	
file?type=png&delay=100&v=7	png	200 ms	Low	
file?type=png&delay=100&v=8	png	202 ms	Low	
file?type=png&delay=100&v=9	png	200 ms	Low	
file?type=png&delay=100&v=10	png	218 ms	Low	

2 HIGH js

10 LOW img

End of
tight mode



Name	Type	Time	Priority	Waterfall
tightmode_simple.html	document	59 ms	Highest	
file?v=1&type=js&delay=2500	script	2.60 s	High	
file?v=2&type=js&delay=2500	script	2.60 s	High	
file?type=png&delay=100&v=1	png	198 ms	Medium	
file?type=png&delay=100&v=2	png	198 ms	Medium	
file?type=png&delay=100&v=3	png	194 ms	Medium	
file?type=png&delay=100&v=4	png	194 ms	Medium	
file?type=png&delay=100&v=5	png	199 ms	Medium	
file?type=png&delay=100&v=6	png	162 ms	Low	
file?type=png&delay=100&v=7	png	158 ms	Low	
file?type=png&delay=100&v=8	png	163 ms	Low	
file?type=png&delay=100&v=9	png	162 ms	Low	
file?type=png&delay=100&v=10	png	162 ms	Low	

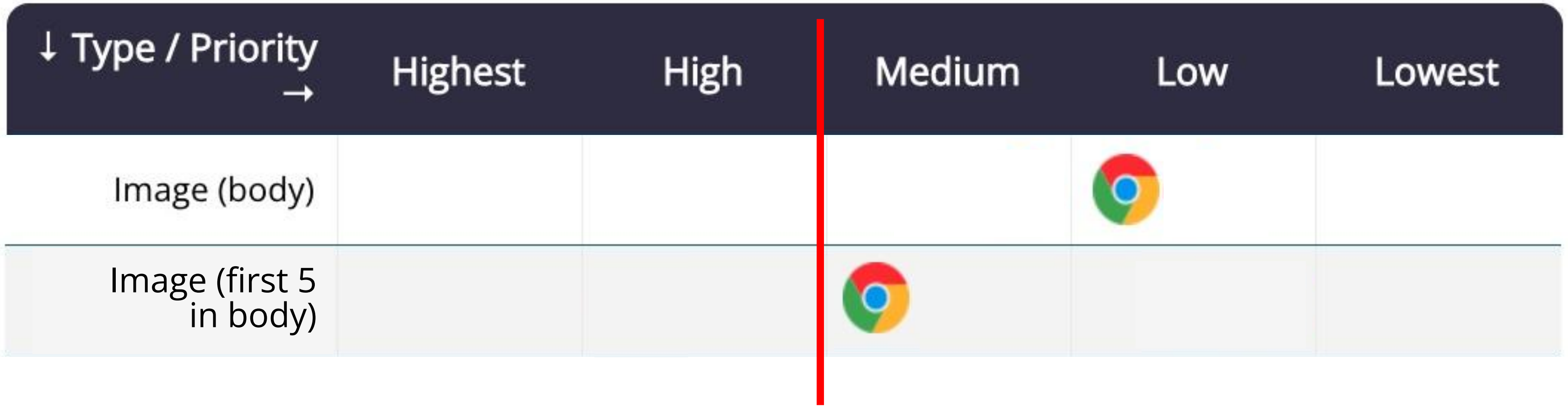
2 HIGH js

5 MEDIUM img

5 LOW img

End of tight mode

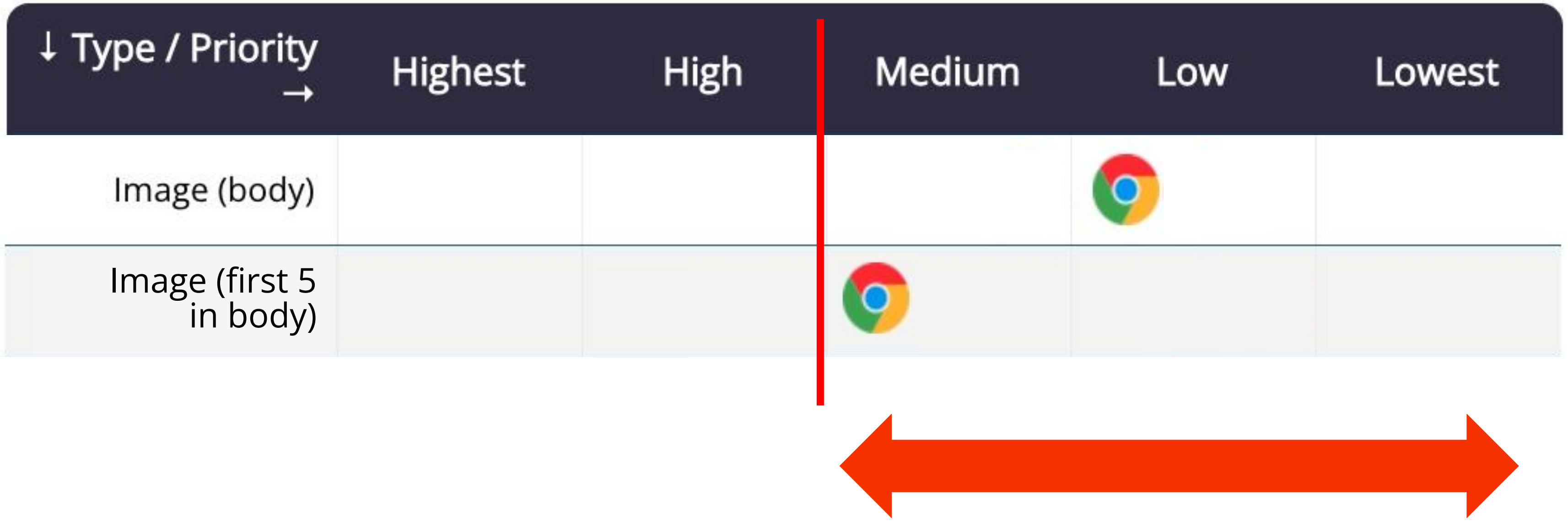
Trying to improve LCP on the entire Web



↓ Type / Priority →	Highest	High	Medium	Low	Lowest
Image (body)				●	
Image (first 5 in body)			●		

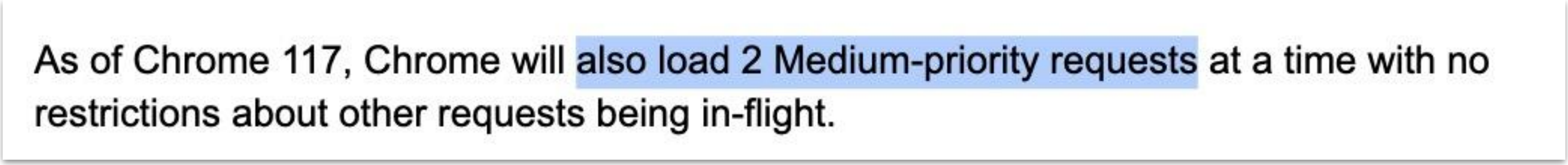
Actively delayed

Trying to improve LCP on the entire Web

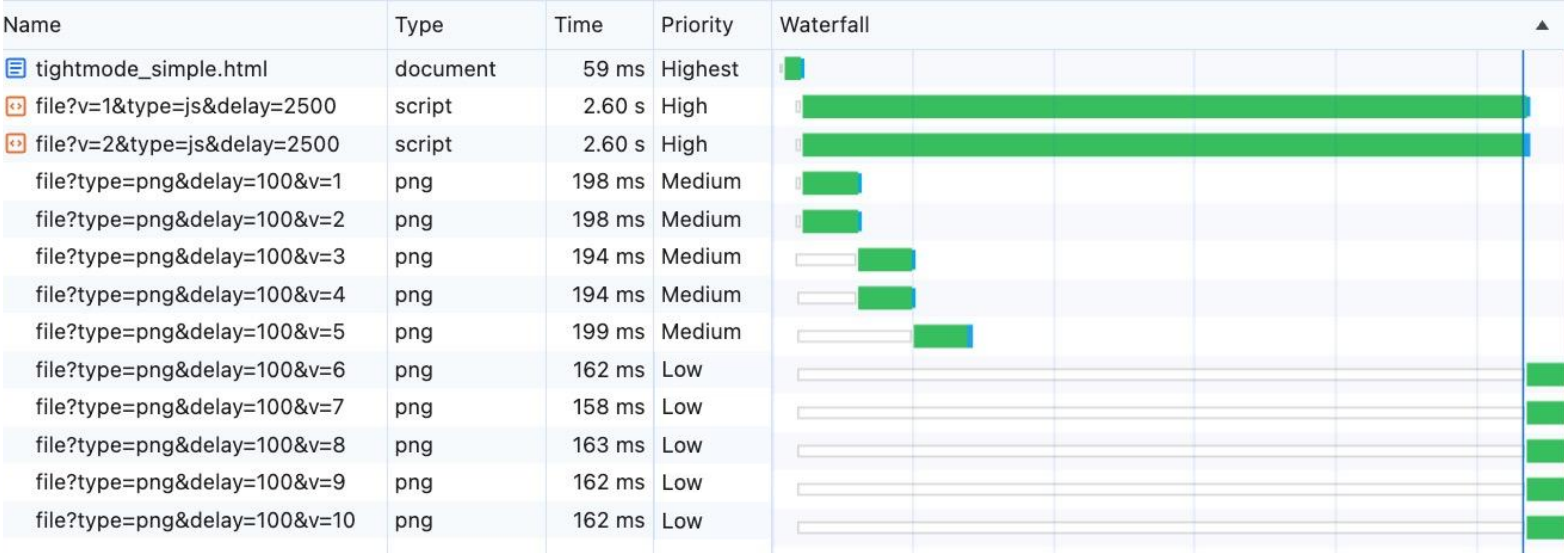


↓ Type / Priority →	Highest	High	Medium	Low	Lowest
Image (body)				●	
Image (first 5 in body)			●		

Actively delayed



As of Chrome 117, Chrome will also load 2 Medium-priority requests at a time with no restrictions about other requests being in-flight.



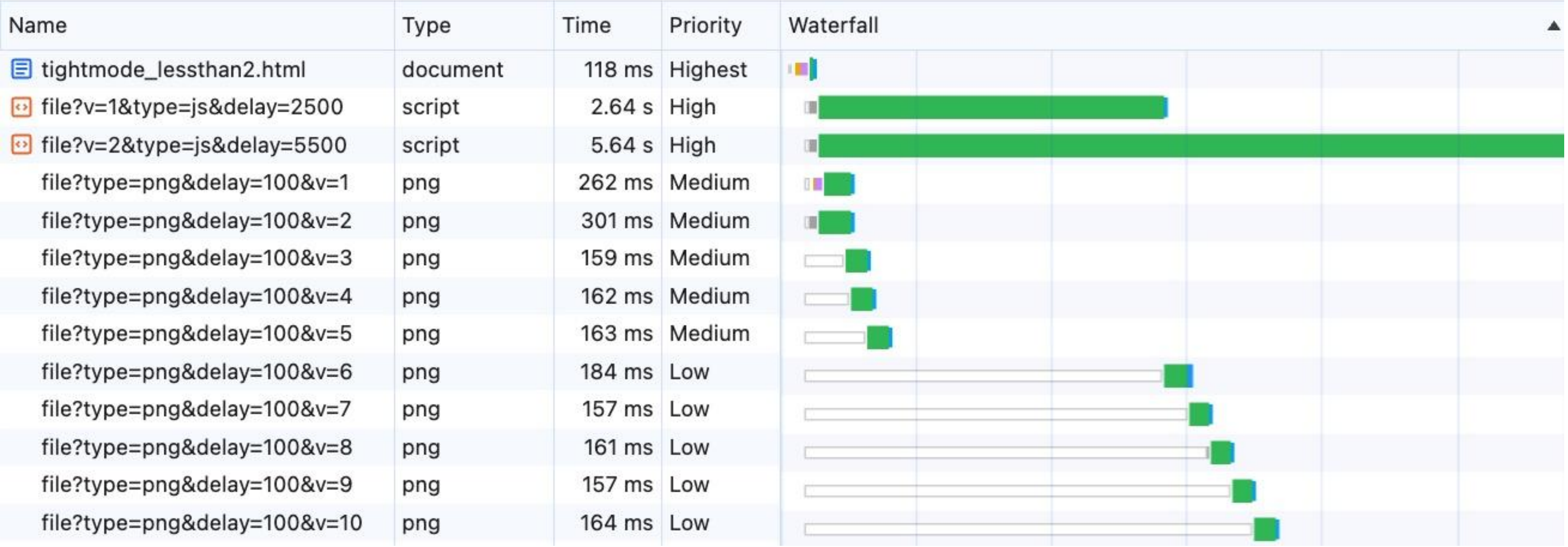
Name	Type	Time	Priority	Waterfall
tightmode_simple.html	document	59 ms	Highest	
file?v=1&type=js&delay=2500	script	2.60 s	High	
file?v=2&type=js&delay=2500	script	2.60 s	High	
file?type=png&delay=100&v=1	png	198 ms	Medium	
file?type=png&delay=100&v=2	png	198 ms	Medium	
file?type=png&delay=100&v=3	png	194 ms	Medium	
file?type=png&delay=100&v=4	png	194 ms	Medium	
file?type=png&delay=100&v=5	png	199 ms	Medium	
file?type=png&delay=100&v=6	png	162 ms	Low	
file?type=png&delay=100&v=7	png	158 ms	Low	
file?type=png&delay=100&v=8	png	163 ms	Low	
file?type=png&delay=100&v=9	png	162 ms	Low	
file?type=png&delay=100&v=10	png	162 ms	Low	

2 HIGH js

5 MEDIUM img

5 LOW img

End of tight mode



Name	Type	Time	Priority	Waterfall
tightmode_lessthan2.html	document	118 ms	Highest	
file?v=1&type=js&delay=2500	script	2.64 s	High	
file?v=2&type=js&delay=5500	script	5.64 s	High	
file?type=png&delay=100&v=1	png	262 ms	Medium	
file?type=png&delay=100&v=2	png	301 ms	Medium	
file?type=png&delay=100&v=3	png	159 ms	Medium	
file?type=png&delay=100&v=4	png	162 ms	Medium	
file?type=png&delay=100&v=5	png	163 ms	Medium	
file?type=png&delay=100&v=6	png	184 ms	Low	
file?type=png&delay=100&v=7	png	157 ms	Low	
file?type=png&delay=100&v=8	png	161 ms	Low	
file?type=png&delay=100&v=9	png	157 ms	Low	
file?type=png&delay=100&v=10	png	164 ms	Low	

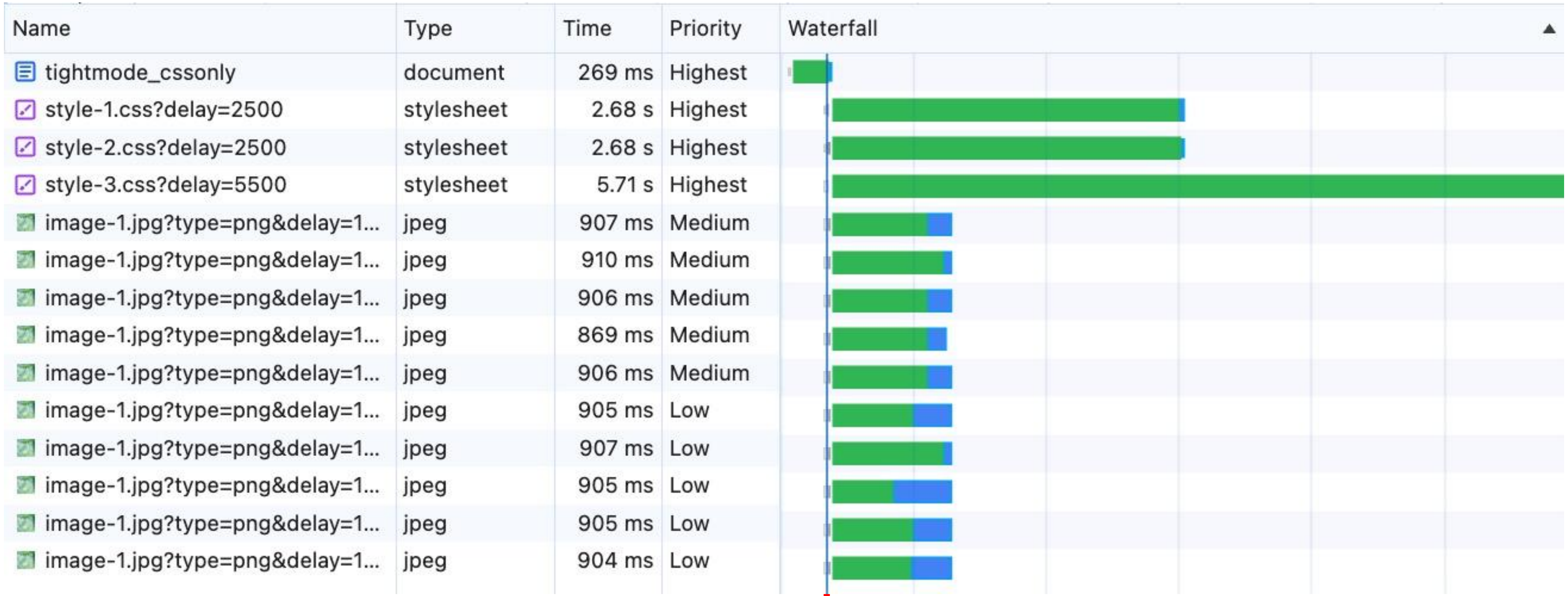
2 HIGH js

5 MEDIUM img

5 LOW img

“Low priority resources are only loaded if there are ***less than 2*** in-flight requests”

End of tight mode



Name	Type	Time	Priority	Waterfall
tightmode_cssonly	document	269 ms	Highest	
style-1.css?delay=2500	stylesheet	2.68 s	Highest	
style-2.css?delay=2500	stylesheet	2.68 s	Highest	
style-3.css?delay=5500	stylesheet	5.71 s	Highest	
image-1.jpg?type=png&delay=1...	jpeg	907 ms	Medium	
image-1.jpg?type=png&delay=1...	jpeg	910 ms	Medium	
image-1.jpg?type=png&delay=1...	jpeg	906 ms	Medium	
image-1.jpg?type=png&delay=1...	jpeg	869 ms	Medium	
image-1.jpg?type=png&delay=1...	jpeg	906 ms	Medium	
image-1.jpg?type=png&delay=1...	jpeg	905 ms	Low	
image-1.jpg?type=png&delay=1...	jpeg	907 ms	Low	
image-1.jpg?type=png&delay=1...	jpeg	905 ms	Low	
image-1.jpg?type=png&delay=1...	jpeg	905 ms	Low	
image-1.jpg?type=png&delay=1...	jpeg	904 ms	Low	

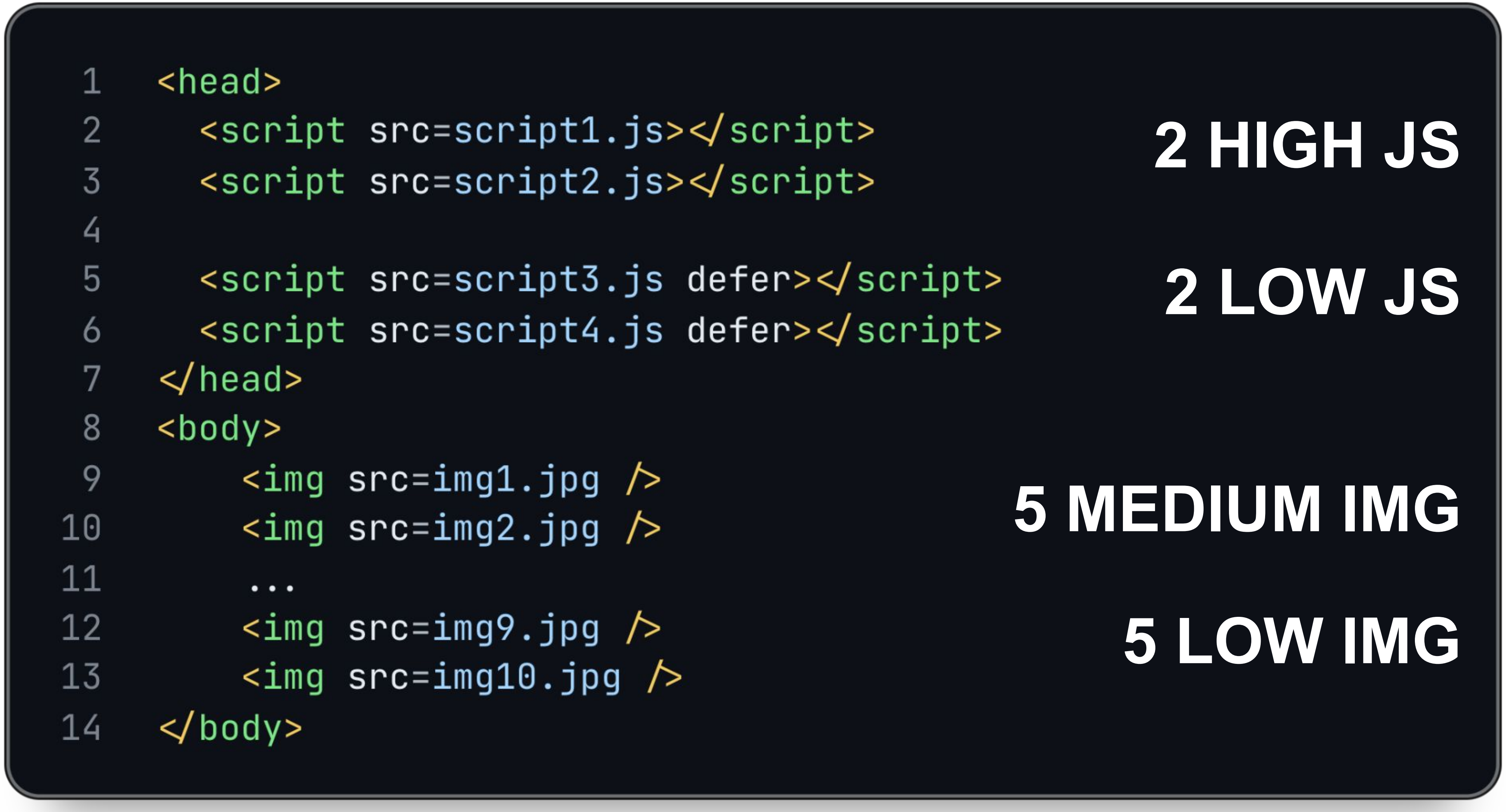
3 HIGHEST CSS

5 MEDIUM img

5 LOW img

End of tight mode

“After all blocking ***scripts*** have been executed”



```
<head>
  <script src=script1.js></script>
  <script src=script2.js></script>

  <script src=script3.js defer></script>
  <script src=script4.js defer></script>
</head>
<body>
    <img src=img1.jpg />
    <img src=img2.jpg />
    ...
    <img src=img9.jpg />
    <img src=img10.jpg />
</body>
```

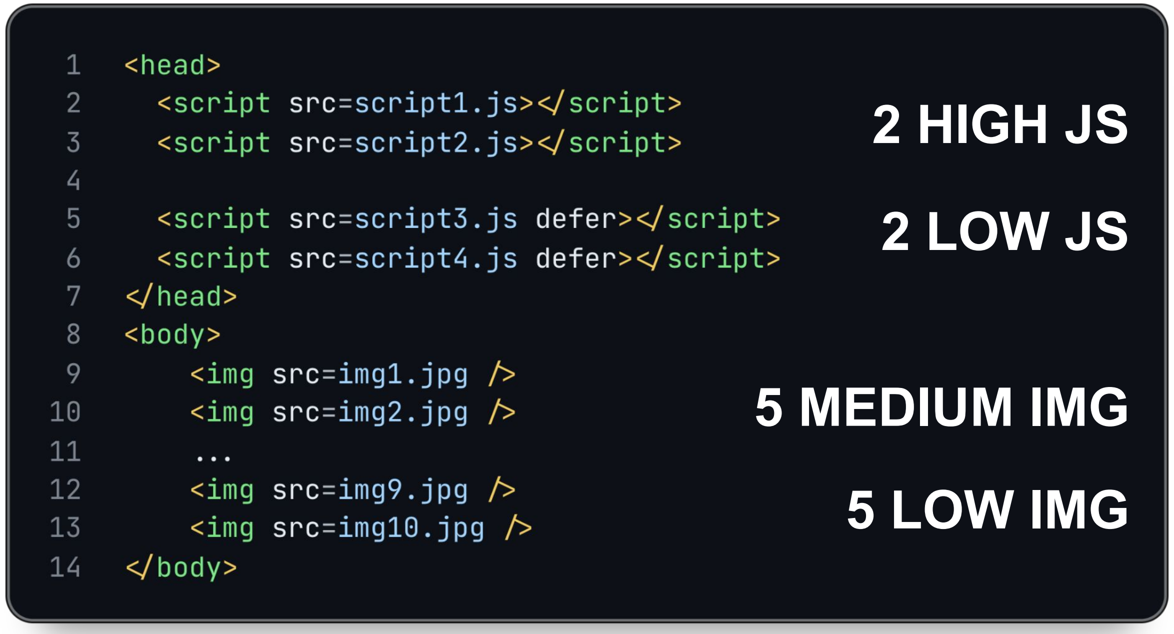
2 HIGH JS

2 LOW JS

5 MEDIUM IMG

5 LOW IMG

What will the waterfall look like for this HTML?



```
<head>
  <script src=script1.js></script>
  <script src=script2.js></script>

  <script src=script3.js defer></script>
  <script src=script4.js defer></script>
</head>
<body>
    <img src=img1.jpg />
    <img src=img2.jpg />
    ...
    <img src=img9.jpg />
    <img src=img10.jpg />
</body>
```

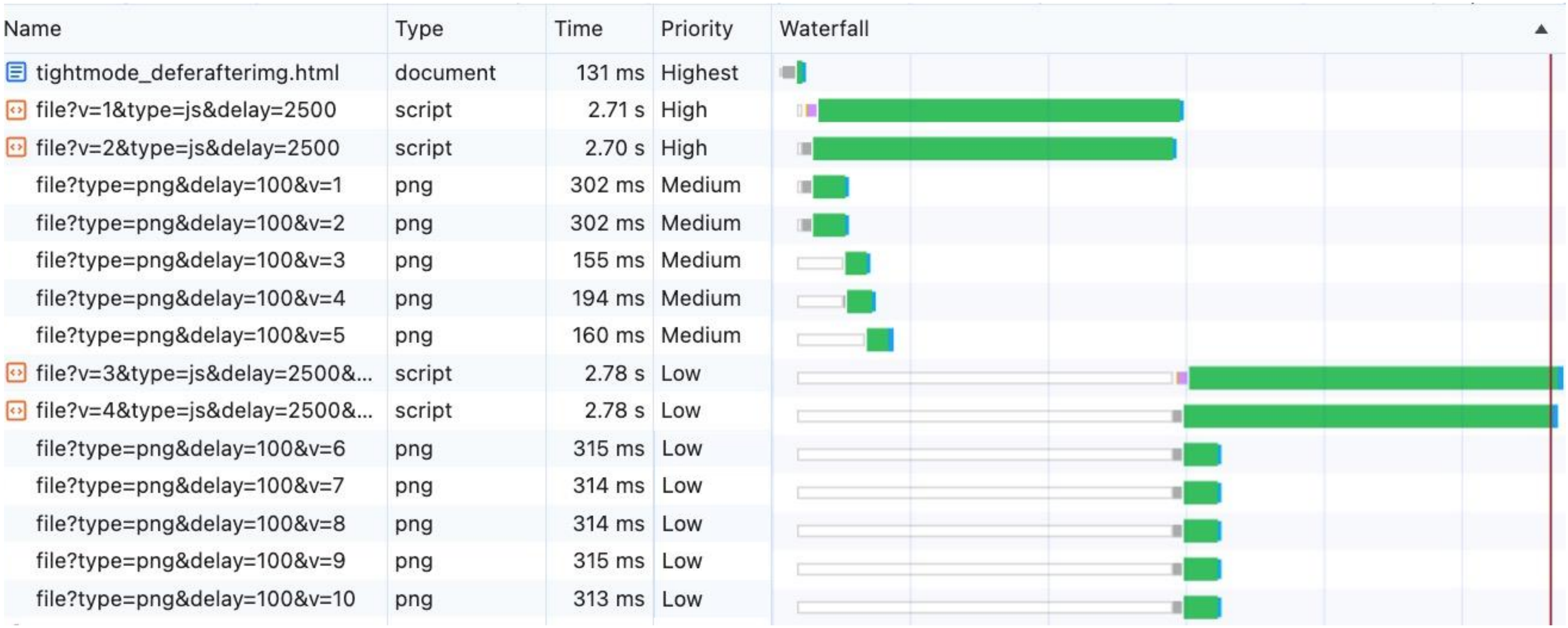
2 HIGH JS

2 LOW JS

5 MEDIUM IMG

5 LOW IMG

images in the <body>
delay
defer JS in the <head>



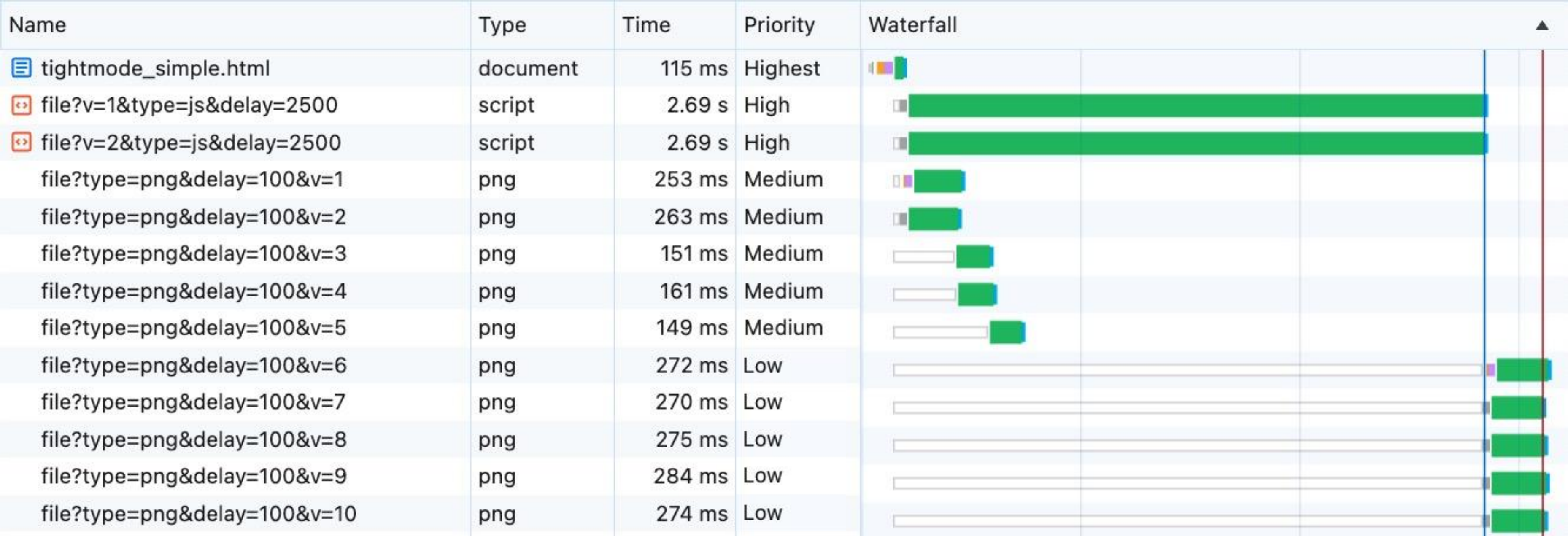
Name	Type	Time	Priority	Waterfall
tightmode_deferafterimg.html	document	131 ms	Highest	
file?v=1&type=js&delay=2500	script	2.71 s	High	
file?v=2&type=js&delay=2500	script	2.70 s	High	
file?type=png&delay=100&v=1	png	302 ms	Medium	
file?type=png&delay=100&v=2	png	302 ms	Medium	
file?type=png&delay=100&v=3	png	155 ms	Medium	
file?type=png&delay=100&v=4	png	194 ms	Medium	
file?type=png&delay=100&v=5	png	160 ms	Medium	
file?v=3&type=js&delay=2500&...	script	2.78 s	Low	
file?v=4&type=js&delay=2500&...	script	2.78 s	Low	
file?type=png&delay=100&v=6	png	315 ms	Low	
file?type=png&delay=100&v=7	png	314 ms	Low	
file?type=png&delay=100&v=8	png	314 ms	Low	
file?type=png&delay=100&v=9	png	315 ms	Low	
file?type=png&delay=100&v=10	png	313 ms	Low	

2 HIGH JS

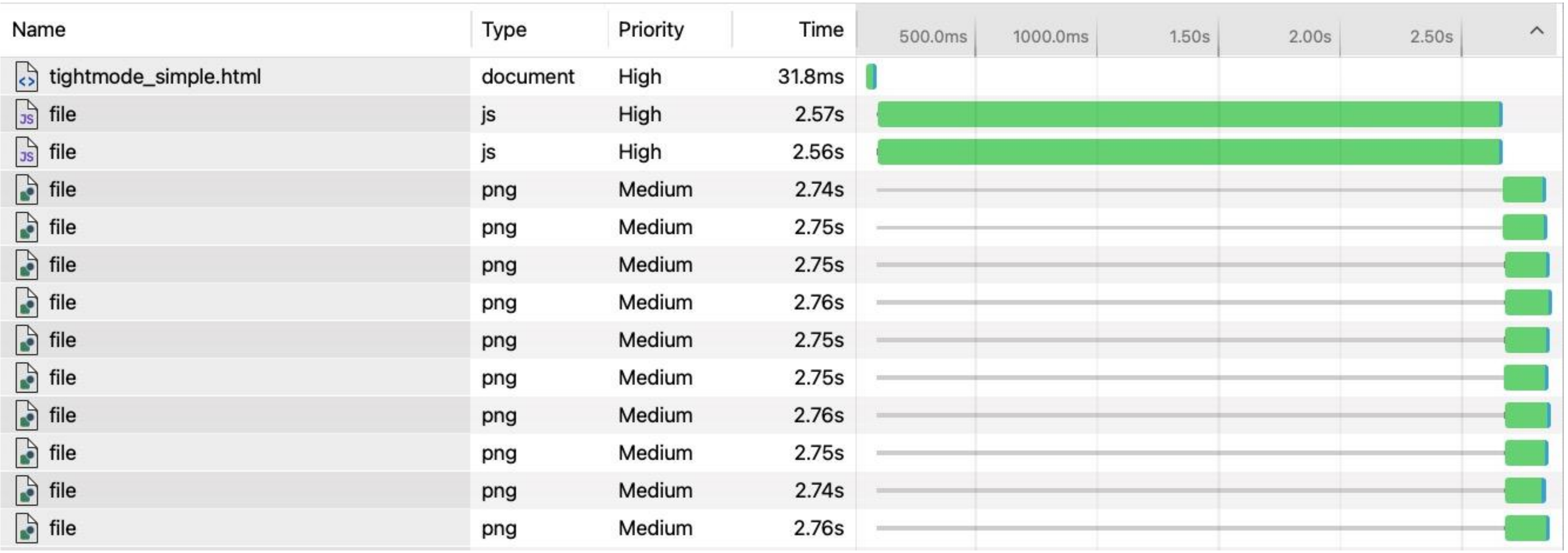
5 MEDIUM IMG

2 LOW JS

5 LOW IMG

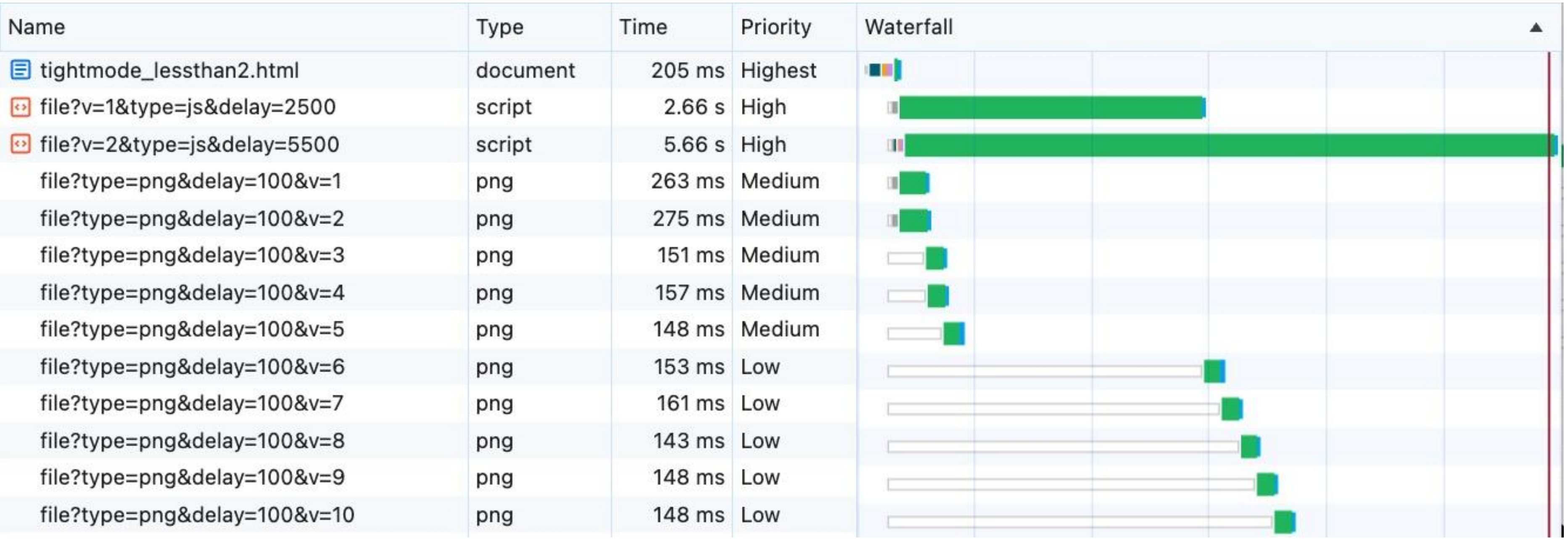


Name	Type	Time	Priority	Waterfall
tightmode_simple.html	document	115 ms	Highest	
file?v=1&type=js&delay=2500	script	2.69 s	High	
file?v=2&type=js&delay=2500	script	2.69 s	High	
file?type=png&delay=100&v=1	png	253 ms	Medium	
file?type=png&delay=100&v=2	png	263 ms	Medium	
file?type=png&delay=100&v=3	png	151 ms	Medium	
file?type=png&delay=100&v=4	png	161 ms	Medium	
file?type=png&delay=100&v=5	png	149 ms	Medium	
file?type=png&delay=100&v=6	png	272 ms	Low	
file?type=png&delay=100&v=7	png	270 ms	Low	
file?type=png&delay=100&v=8	png	275 ms	Low	
file?type=png&delay=100&v=9	png	284 ms	Low	
file?type=png&delay=100&v=10	png	274 ms	Low	

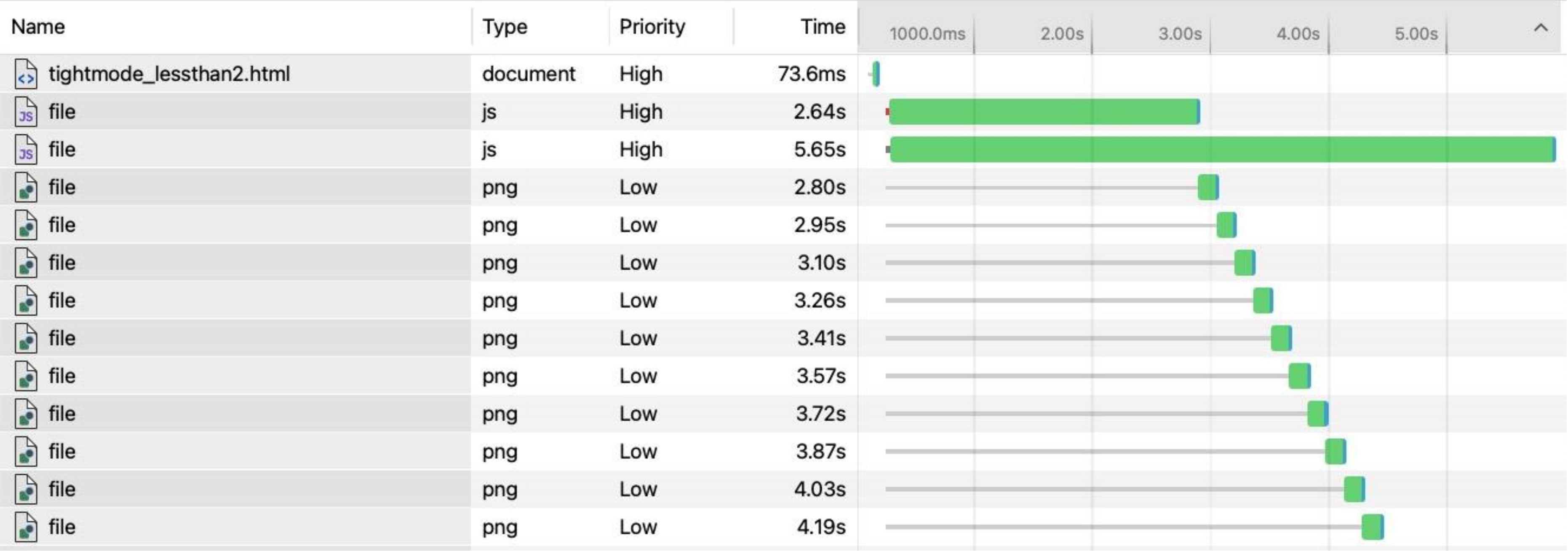


Name	Type	Priority	Time	
tightmode_simple.html	document	High	31.8ms	
file	js	High	2.57s	
file	js	High	2.56s	
file	png	Medium	2.74s	
file	png	Medium	2.75s	
file	png	Medium	2.75s	
file	png	Medium	2.76s	
file	png	Medium	2.75s	
file	png	Medium	2.75s	
file	png	Medium	2.76s	
file	png	Medium	2.75s	
file	png	Medium	2.74s	
file	png	Medium	2.76s	

No special casing of first 5 images

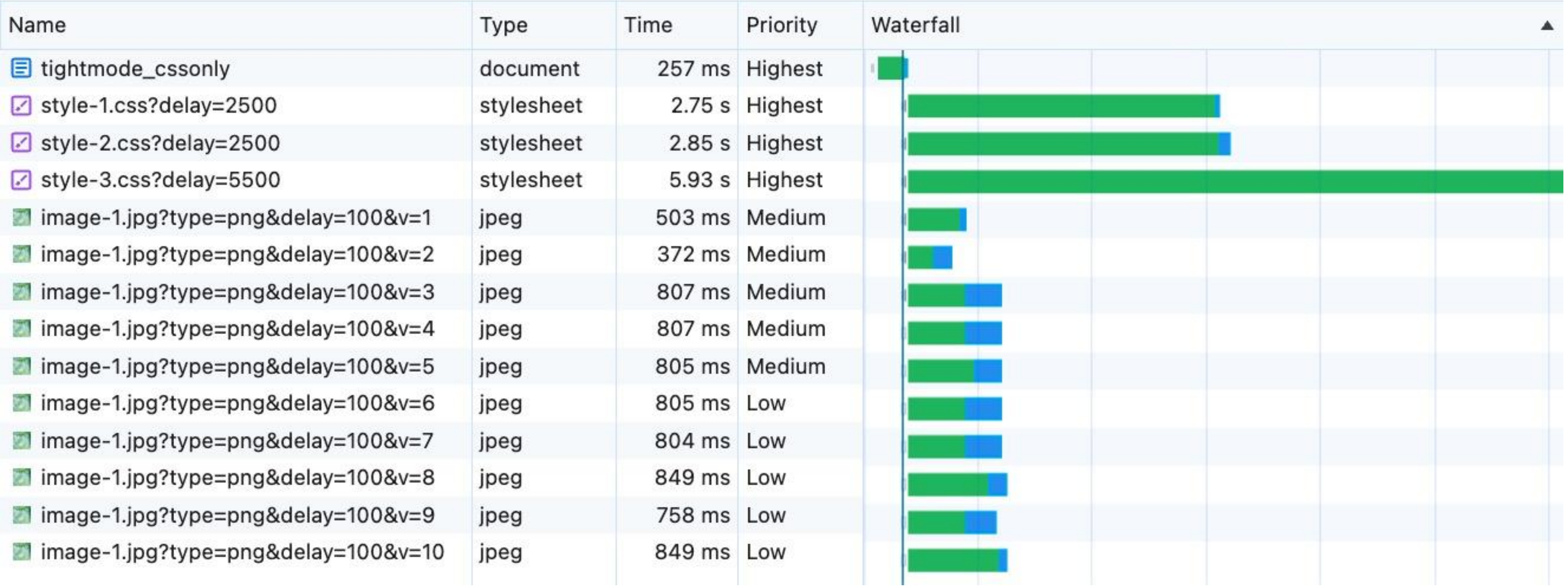


Name	Type	Time	Priority	Waterfall
tightmode_lessthan2.html	document	205 ms	Highest	
file?v=1&type=js&delay=2500	script	2.66 s	High	
file?v=2&type=js&delay=5500	script	5.66 s	High	
file?type=png&delay=100&v=1	png	263 ms	Medium	
file?type=png&delay=100&v=2	png	275 ms	Medium	
file?type=png&delay=100&v=3	png	151 ms	Medium	
file?type=png&delay=100&v=4	png	157 ms	Medium	
file?type=png&delay=100&v=5	png	148 ms	Medium	
file?type=png&delay=100&v=6	png	153 ms	Low	
file?type=png&delay=100&v=7	png	161 ms	Low	
file?type=png&delay=100&v=8	png	143 ms	Low	
file?type=png&delay=100&v=9	png	148 ms	Low	
file?type=png&delay=100&v=10	png	148 ms	Low	

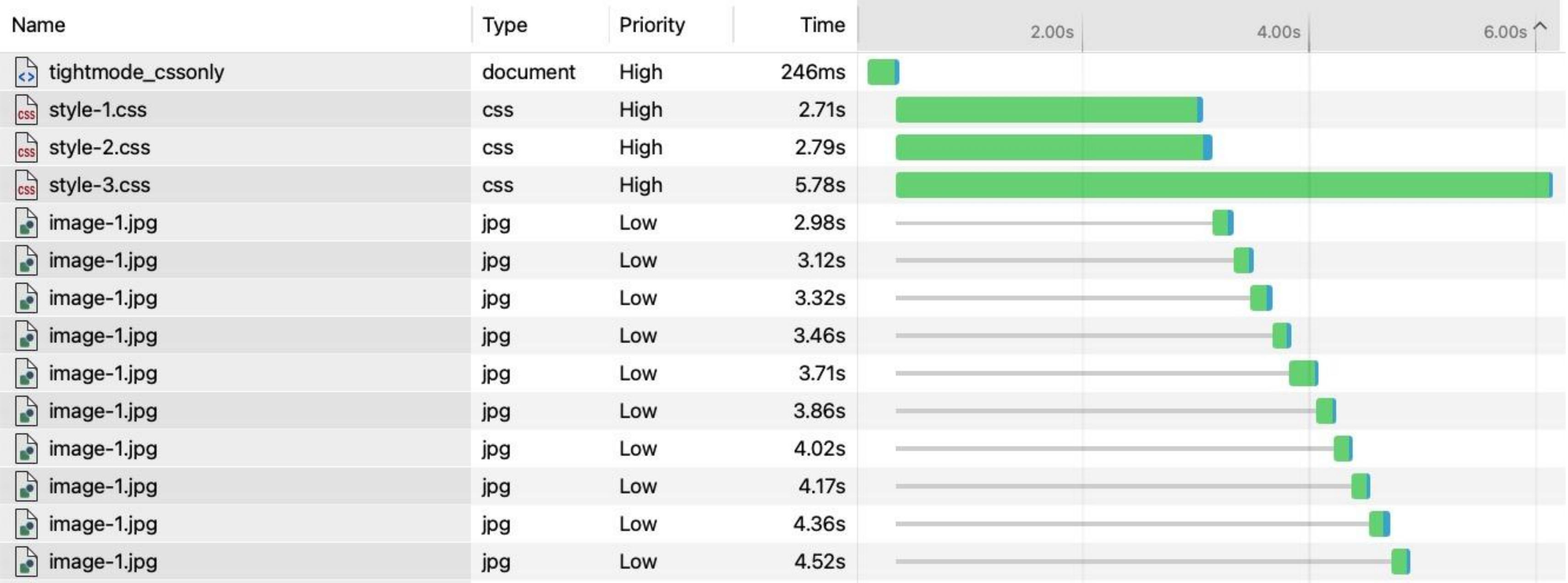


Name	Type	Priority	Time
tightmode_lessthan2.html	document	High	73.6ms
file	js	High	2.64s
file	js	High	5.65s
file	png	Low	2.80s
file	png	Low	2.95s
file	png	Low	3.10s
file	png	Low	3.26s
file	png	Low	3.41s
file	png	Low	3.57s
file	png	Low	3.72s
file	png	Low	3.87s
file	png	Low	4.03s
file	png	Low	4.19s

Max 2 things in flight



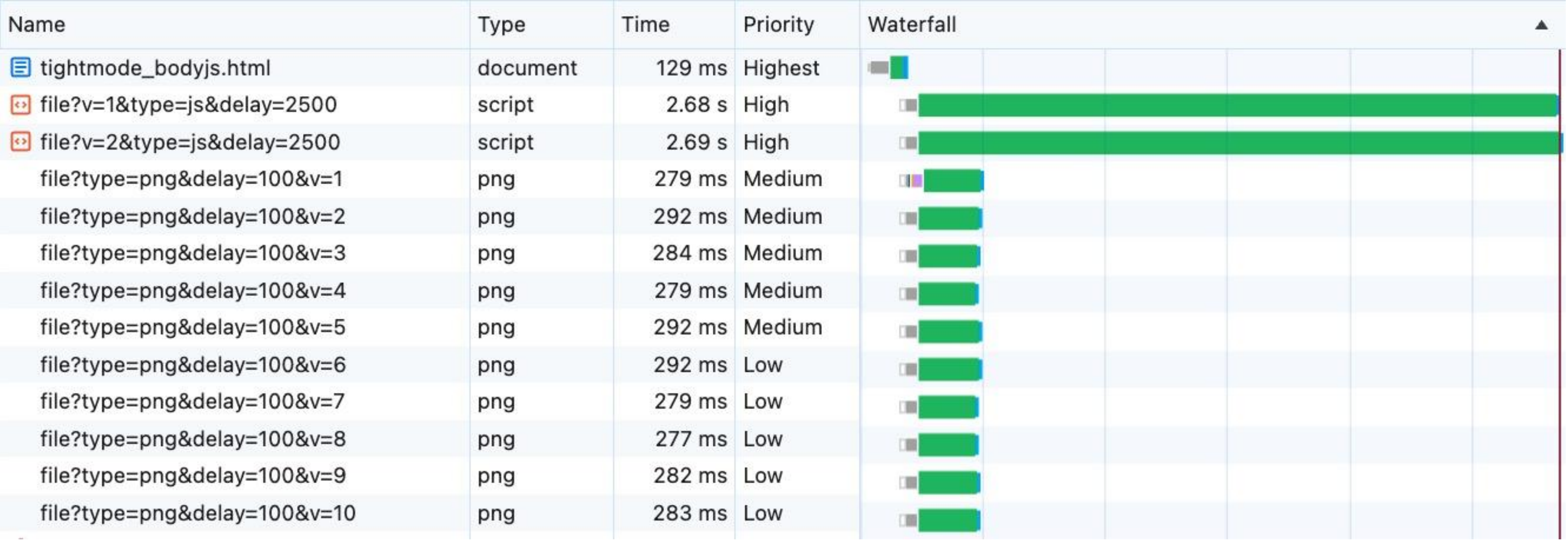
Name	Type	Time	Priority	Waterfall
tightmode_cssonly	document	257 ms	Highest	
style-1.css?delay=2500	stylesheet	2.75 s	Highest	
style-2.css?delay=2500	stylesheet	2.85 s	Highest	
style-3.css?delay=5500	stylesheet	5.93 s	Highest	
image-1.jpg?type=png&delay=100&v=1	jpeg	503 ms	Medium	
image-1.jpg?type=png&delay=100&v=2	jpeg	372 ms	Medium	
image-1.jpg?type=png&delay=100&v=3	jpeg	807 ms	Medium	
image-1.jpg?type=png&delay=100&v=4	jpeg	807 ms	Medium	
image-1.jpg?type=png&delay=100&v=5	jpeg	805 ms	Medium	
image-1.jpg?type=png&delay=100&v=6	jpeg	805 ms	Low	
image-1.jpg?type=png&delay=100&v=7	jpeg	804 ms	Low	
image-1.jpg?type=png&delay=100&v=8	jpeg	849 ms	Low	
image-1.jpg?type=png&delay=100&v=9	jpeg	758 ms	Low	
image-1.jpg?type=png&delay=100&v=10	jpeg	849 ms	Low	



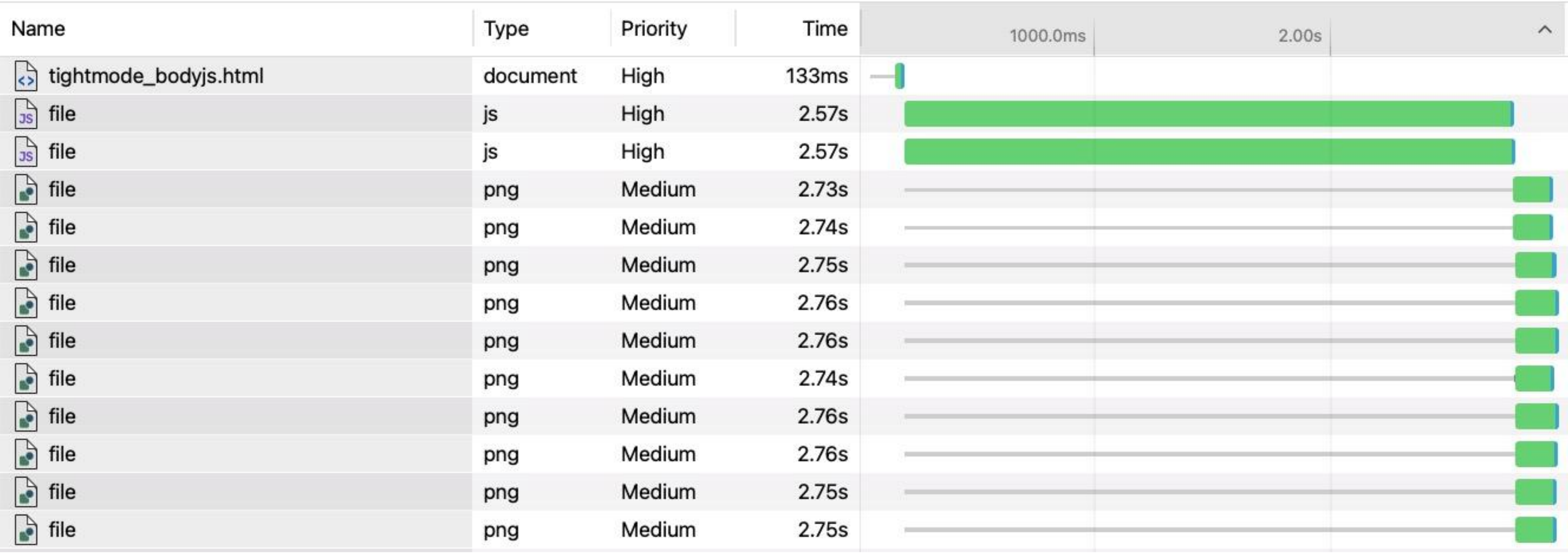
Name	Type	Priority	Time	2.00s 4.00s 6.00s
tightmode_cssonly	document	High	246ms	
style-1.css	css	High	2.71s	
style-2.css	css	High	2.79s	
style-3.css	css	High	5.78s	
image-1.jpg	jpg	Low	2.98s	
image-1.jpg	jpg	Low	3.12s	
image-1.jpg	jpg	Low	3.32s	
image-1.jpg	jpg	Low	3.46s	
image-1.jpg	jpg	Low	3.71s	
image-1.jpg	jpg	Low	3.86s	
image-1.jpg	jpg	Low	4.02s	
image-1.jpg	jpg	Low	4.17s	
image-1.jpg	jpg	Low	4.36s	
image-1.jpg	jpg	Low	4.52s	

CSS also triggers tight mode!

JS top of <body>



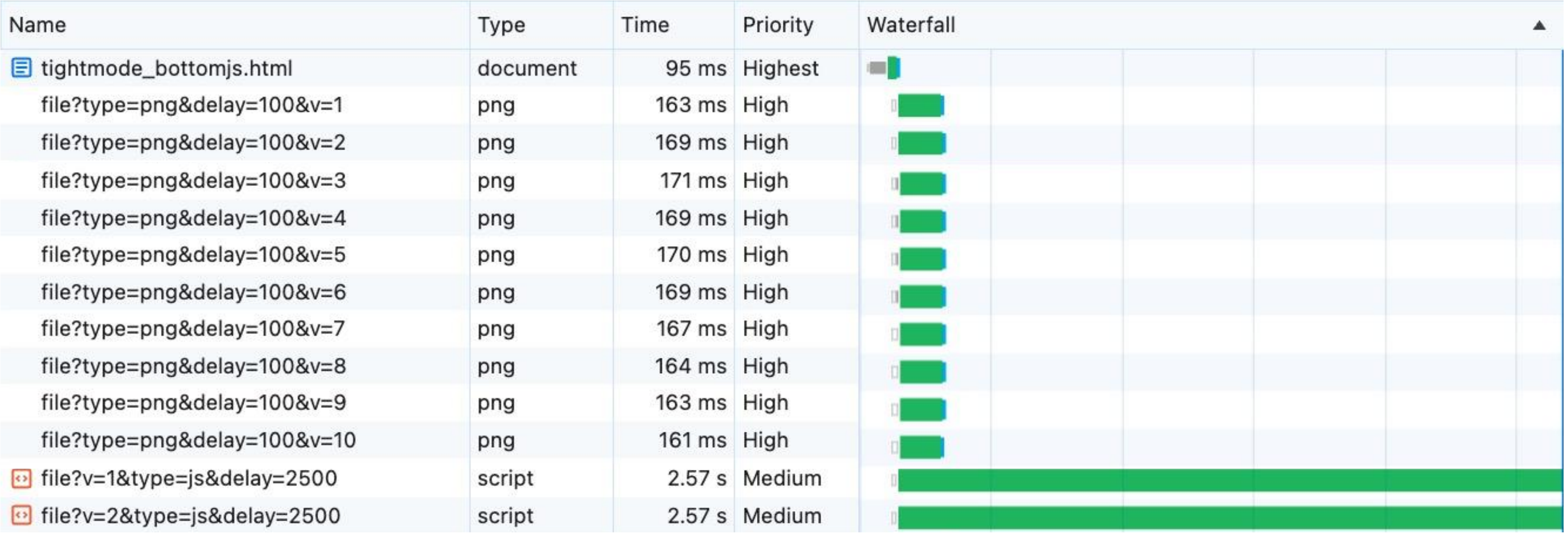
Name	Type	Time	Priority	Waterfall
tightmode_bodyjs.html	document	129 ms	Highest	
file?v=1&type=js&delay=2500	script	2.68 s	High	
file?v=2&type=js&delay=2500	script	2.69 s	High	
file?type=png&delay=100&v=1	png	279 ms	Medium	
file?type=png&delay=100&v=2	png	292 ms	Medium	
file?type=png&delay=100&v=3	png	284 ms	Medium	
file?type=png&delay=100&v=4	png	279 ms	Medium	
file?type=png&delay=100&v=5	png	292 ms	Medium	
file?type=png&delay=100&v=6	png	292 ms	Low	
file?type=png&delay=100&v=7	png	279 ms	Low	
file?type=png&delay=100&v=8	png	277 ms	Low	
file?type=png&delay=100&v=9	png	282 ms	Low	
file?type=png&delay=100&v=10	png	283 ms	Low	



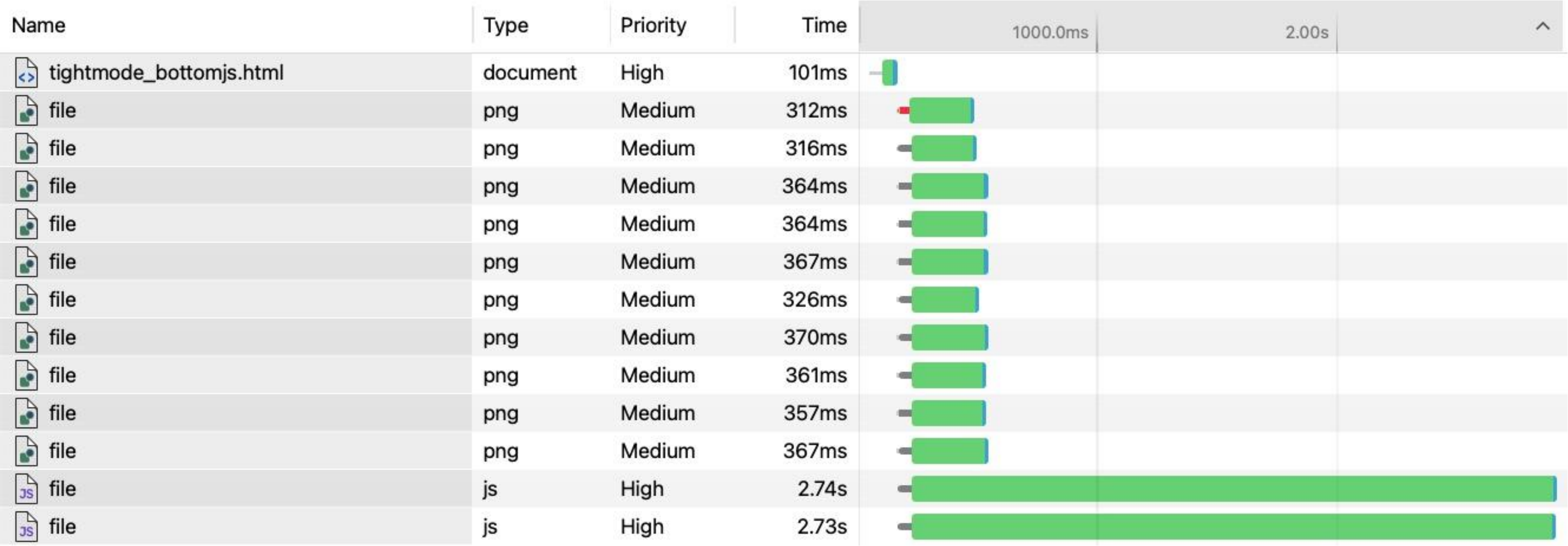
Name	Type	Priority	Time	1000.0ms 2.00s
tightmode_bodyjs.html	document	High	133ms	
file	js	High	2.57s	
file	js	High	2.57s	
file	png	Medium	2.73s	
file	png	Medium	2.74s	
file	png	Medium	2.75s	
file	png	Medium	2.76s	
file	png	Medium	2.76s	
file	png	Medium	2.74s	
file	png	Medium	2.76s	
file	png	Medium	2.76s	
file	png	Medium	2.75s	
file	png	Medium	2.75s	

Blocking JS or CSS delay whatever's behind them

JS bottom of <body>



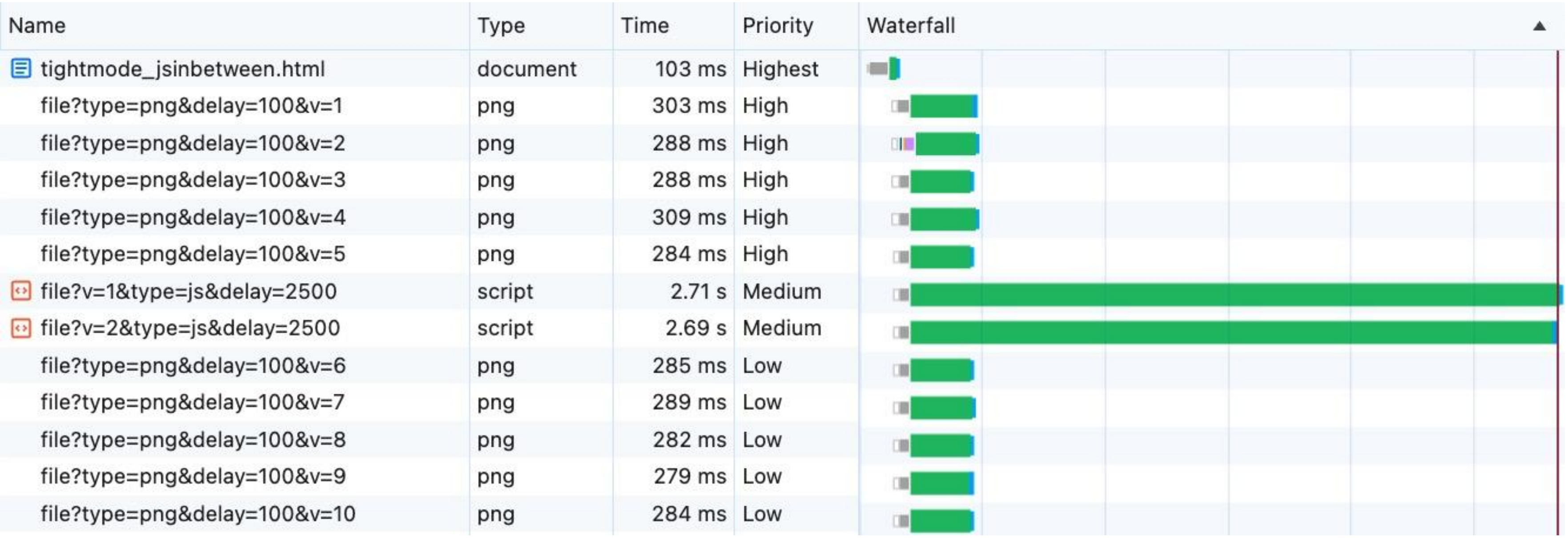
Name	Type	Time	Priority	Waterfall
tightmode_bottomjs.html	document	95 ms	Highest	
file?type=png&delay=100&v=1	png	163 ms	High	
file?type=png&delay=100&v=2	png	169 ms	High	
file?type=png&delay=100&v=3	png	171 ms	High	
file?type=png&delay=100&v=4	png	169 ms	High	
file?type=png&delay=100&v=5	png	170 ms	High	
file?type=png&delay=100&v=6	png	169 ms	High	
file?type=png&delay=100&v=7	png	167 ms	High	
file?type=png&delay=100&v=8	png	164 ms	High	
file?type=png&delay=100&v=9	png	163 ms	High	
file?type=png&delay=100&v=10	png	161 ms	High	
file?v=1&type=js&delay=2500	script	2.57 s	Medium	
file?v=2&type=js&delay=2500	script	2.57 s	Medium	



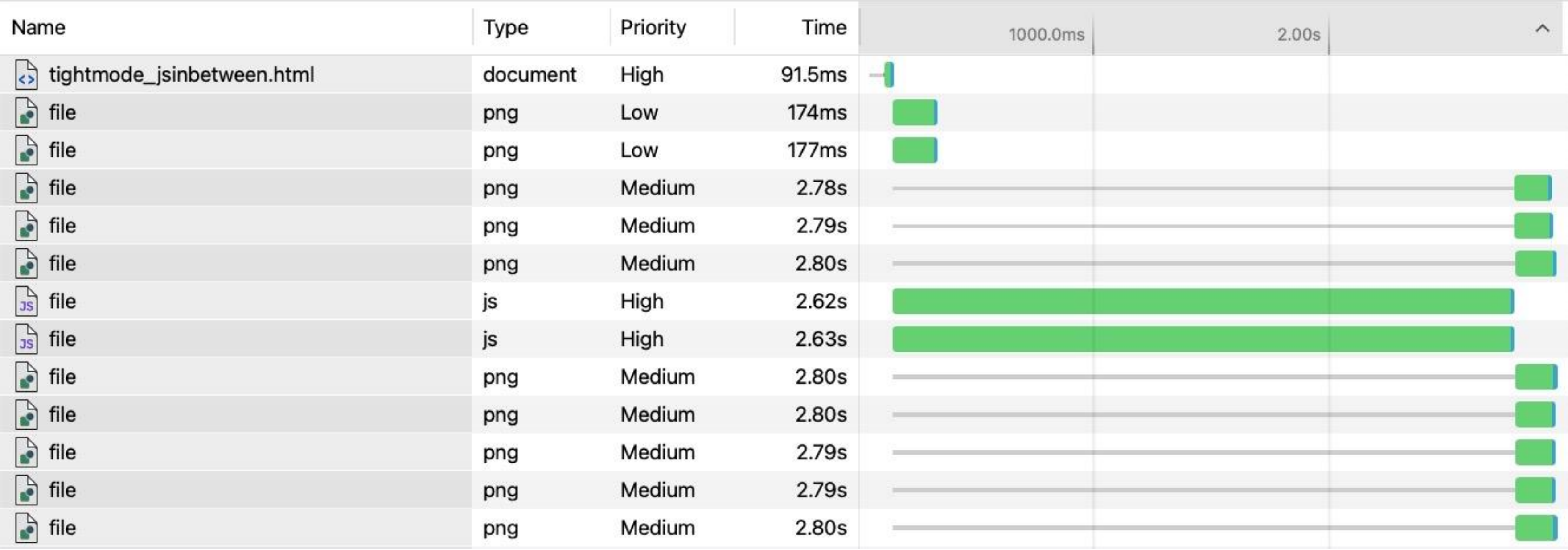
Name	Type	Priority	Time
tightmode_bottomjs.html	document	High	101ms
file	png	Medium	312ms
file	png	Medium	316ms
file	png	Medium	364ms
file	png	Medium	364ms
file	png	Medium	367ms
file	png	Medium	326ms
file	png	Medium	370ms
file	png	Medium	361ms
file	png	Medium	357ms
file	png	Medium	367ms
file	js	High	2.74s
file	js	High	2.73s

Blocking JS or CSS delay whatever's behind them

JS middle of <body>

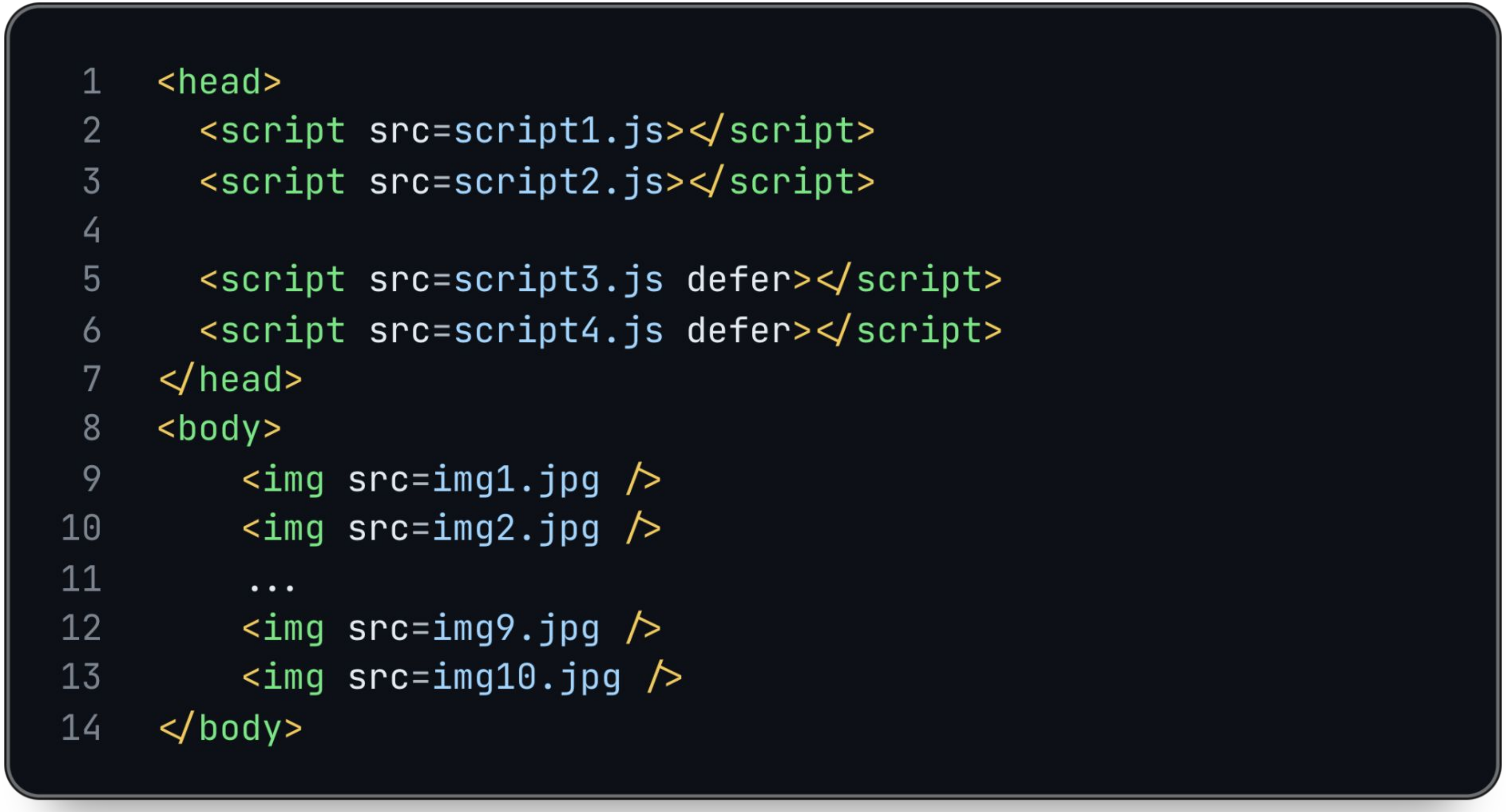


Name	Type	Time	Priority	Waterfall
tightmode_jsinbetween.html	document	103 ms	Highest	
file?type=png&delay=100&v=1	png	303 ms	High	
file?type=png&delay=100&v=2	png	288 ms	High	
file?type=png&delay=100&v=3	png	288 ms	High	
file?type=png&delay=100&v=4	png	309 ms	High	
file?type=png&delay=100&v=5	png	284 ms	High	
file?v=1&type=js&delay=2500	script	2.71 s	Medium	
file?v=2&type=js&delay=2500	script	2.69 s	Medium	
file?type=png&delay=100&v=6	png	285 ms	Low	
file?type=png&delay=100&v=7	png	289 ms	Low	
file?type=png&delay=100&v=8	png	282 ms	Low	
file?type=png&delay=100&v=9	png	279 ms	Low	
file?type=png&delay=100&v=10	png	284 ms	Low	



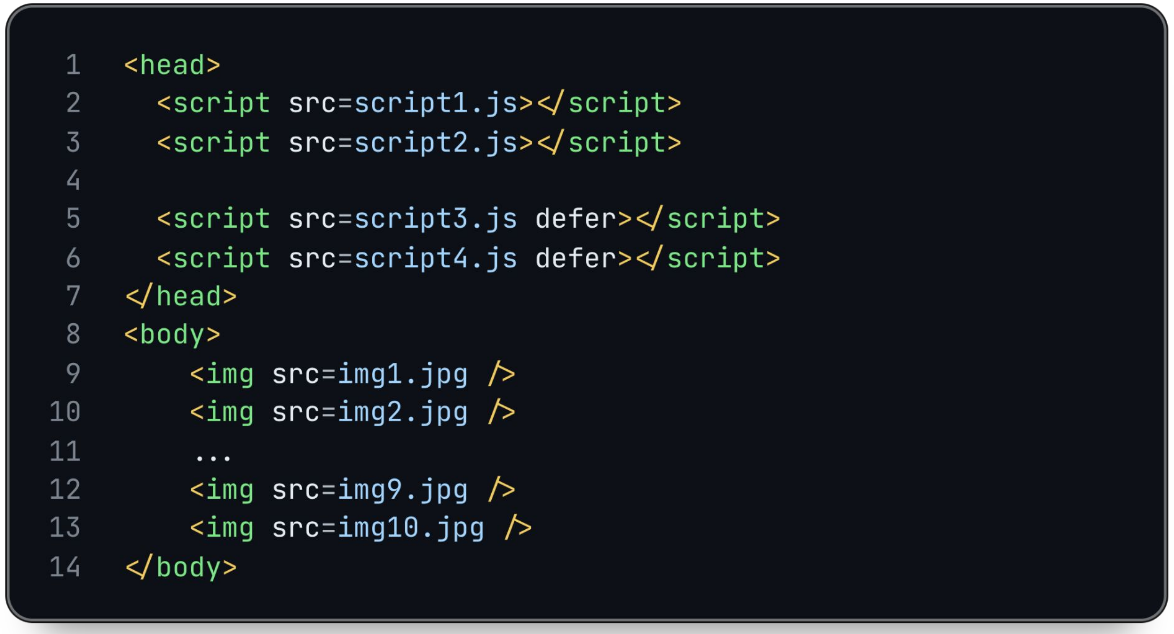
Name	Type	Priority	Time
tightmode_jsinbetween.html	document	High	91.5ms
file	png	Low	174ms
file	png	Low	177ms
file	png	Medium	2.78s
file	png	Medium	2.79s
file	png	Medium	2.80s
file	js	High	2.62s
file	js	High	2.63s
file	png	Medium	2.80s
file	png	Medium	2.80s
file	png	Medium	2.79s
file	png	Medium	2.79s
file	png	Medium	2.80s

Some weird heuristics at work here…

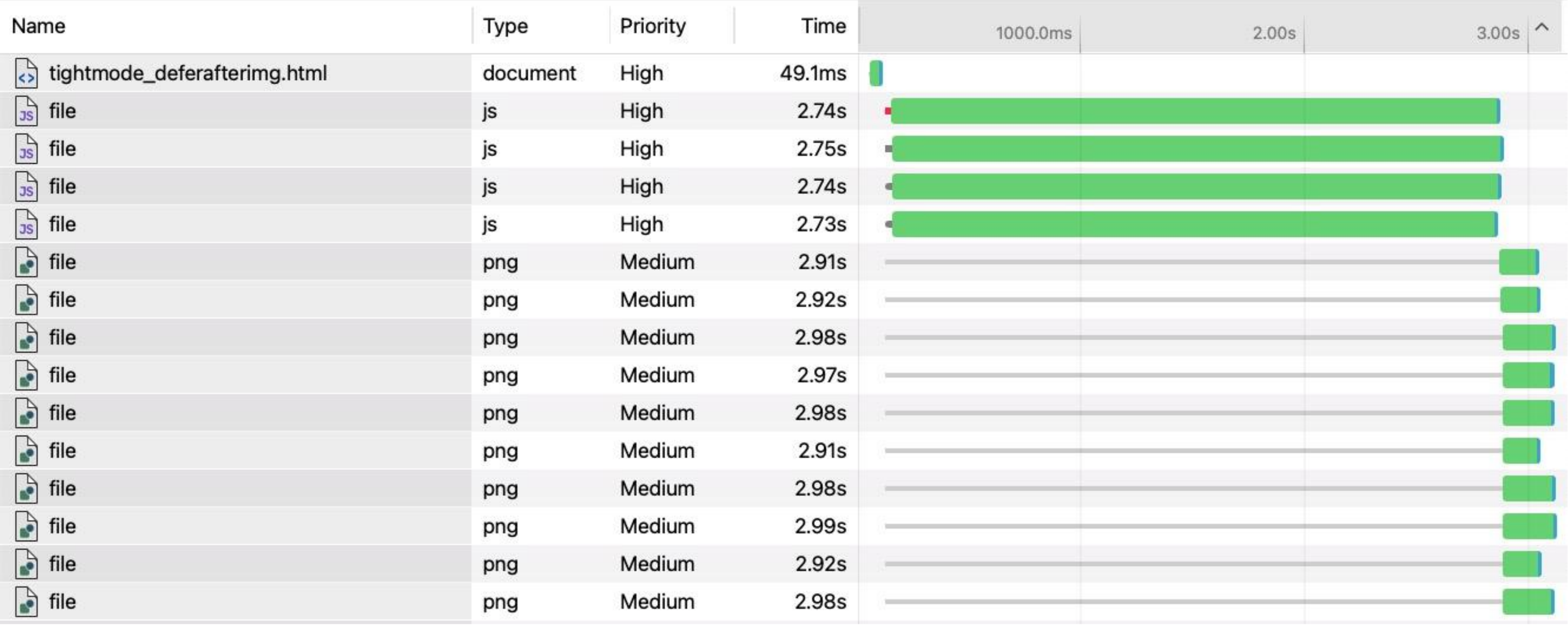


```html
<head>
  <script src=script1.js></script>
  <script src=script2.js></script>

  <script src=script3.js defer></script>
  <script src=script4.js defer></script>
</head>
<body>
    <img src=img1.jpg />
    <img src=img2.jpg />
    ...
    <img src=img9.jpg />
    <img src=img10.jpg />
</body>
```

What will the waterfall look like for this HTML?

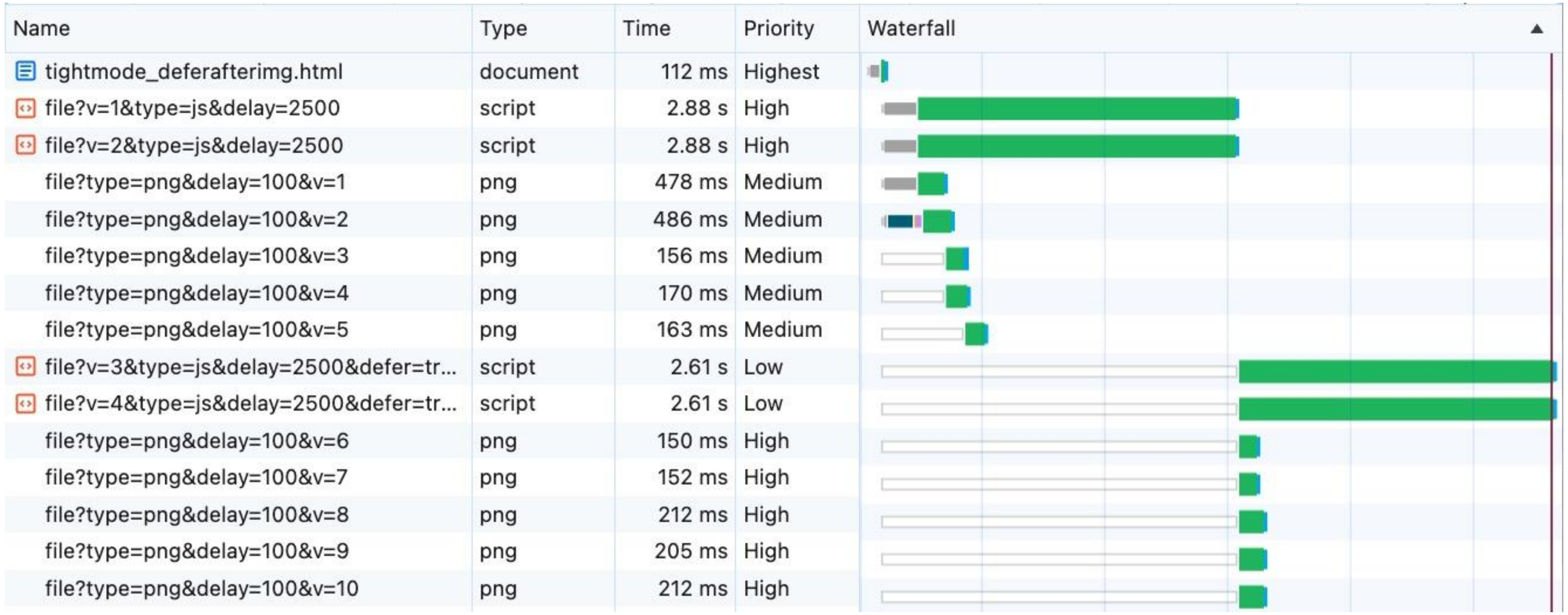


```html
<head>
  <script src=script1.js></script>
  <script src=script2.js></script>

  <script src=script3.js defer></script>
  <script src=script4.js defer></script>
</head>
<body>
    <img src=img1.jpg />
    <img src=img2.jpg />
    ...
    <img src=img9.jpg />
    <img src=img10.jpg />
</body>
```

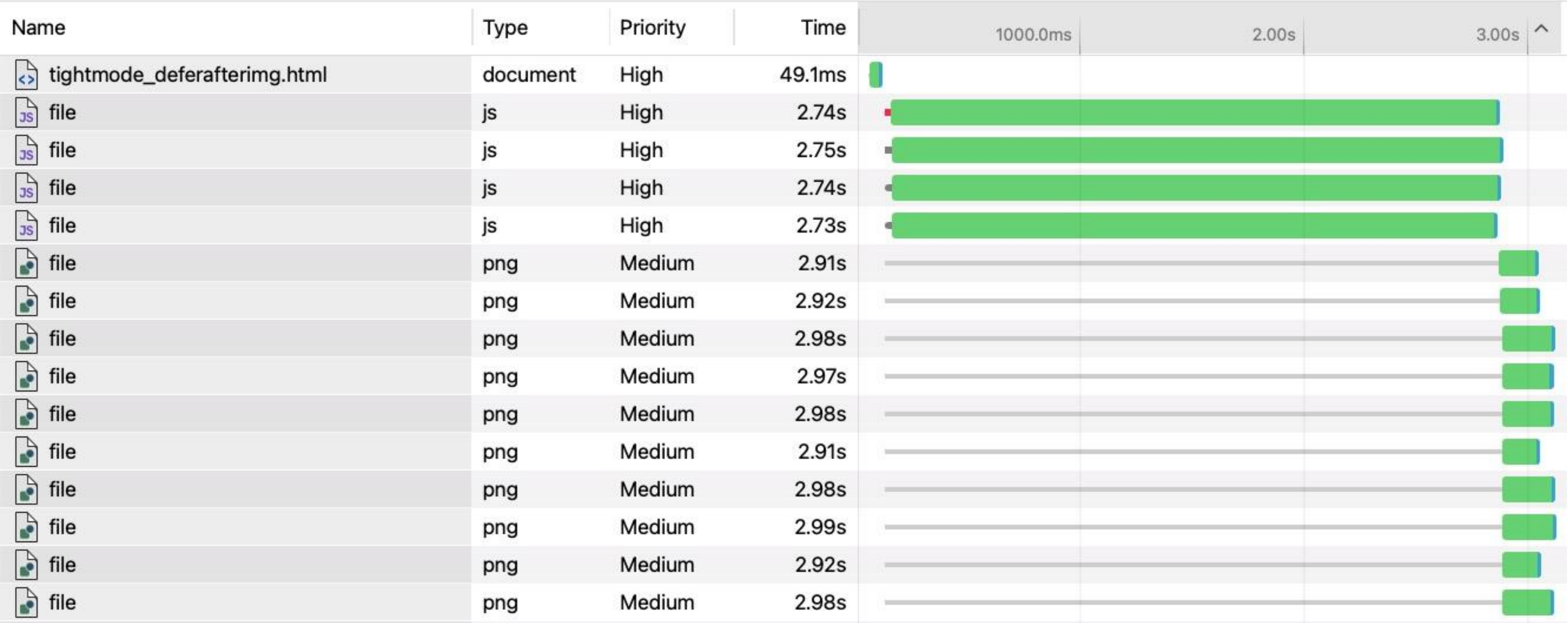


Name	Type	Priority	Time
tightmode_deferafterimg.html	document	High	49.1ms
file	js	High	2.74s
file	js	High	2.75s
file	js	High	2.74s
file	js	High	2.73s
file	png	Medium	2.91s
file	png	Medium	2.92s
file	png	Medium	2.98s
file	png	Medium	2.97s
file	png	Medium	2.98s
file	png	Medium	2.91s
file	png	Medium	2.98s
file	png	Medium	2.99s
file	png	Medium	2.92s
file	png	Medium	2.98s

Async/Defer JS don’t trigger tight mode by themselves, but are downloaded in it



Name	Type	Time	Priority	Waterfall
tightmode_deferafterimg.html	document	112 ms	Highest	
file?v=1&type=js&delay=2500	script	2.88 s	High	
file?v=2&type=js&delay=2500	script	2.88 s	High	
file?type=png&delay=100&v=1	png	478 ms	Medium	
file?type=png&delay=100&v=2	png	486 ms	Medium	
file?type=png&delay=100&v=3	png	156 ms	Medium	
file?type=png&delay=100&v=4	png	170 ms	Medium	
file?type=png&delay=100&v=5	png	163 ms	Medium	
file?v=3&type=js&delay=2500&defer=tr...	script	2.61 s	Low	
file?v=4&type=js&delay=2500&defer=tr...	script	2.61 s	Low	
file?type=png&delay=100&v=6	png	150 ms	High	
file?type=png&delay=100&v=7	png	152 ms	High	
file?type=png&delay=100&v=8	png	212 ms	High	
file?type=png&delay=100&v=9	png	205 ms	High	
file?type=png&delay=100&v=10	png	212 ms	High	



Name	Type	Priority	Time	1000.0ms 2.00s 3.00s
tightmode_deferafterimg.html	document	High	49.1ms	
file	js	High	2.74s	
file	js	High	2.75s	
file	js	High	2.74s	
file	js	High	2.73s	
file	png	Medium	2.91s	
file	png	Medium	2.92s	
file	png	Medium	2.98s	
file	png	Medium	2.97s	
file	png	Medium	2.98s	
file	png	Medium	2.91s	
file	png	Medium	2.98s	
file	png	Medium	2.99s	
file	png	Medium	2.92s	
file	png	Medium	2.98s	

Exact same HTML,

***radically* different behaviour**

Tight mode

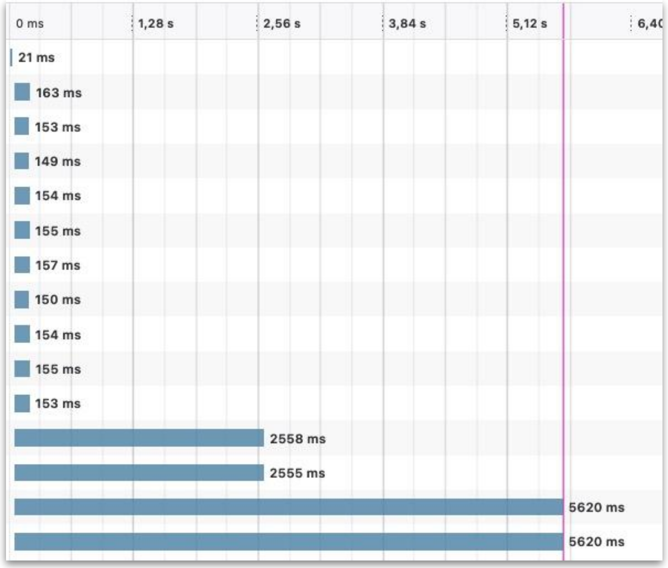
While blocking JS in the <head> is busy

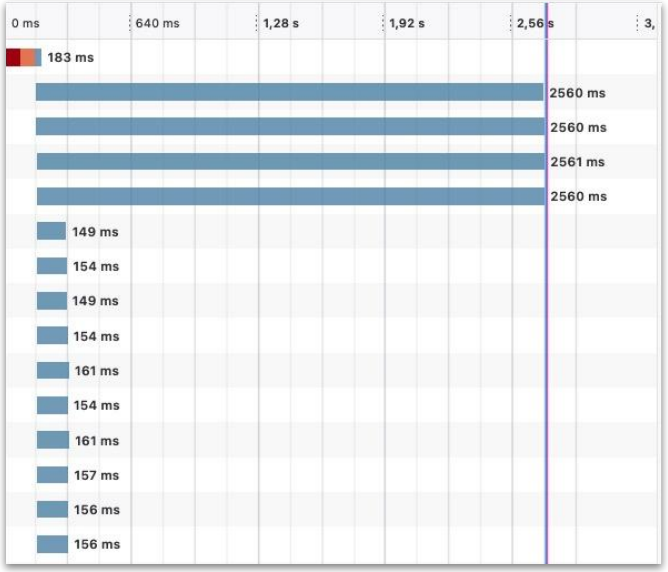
- Only LOW/LOWEST if fewer than 2 things in flight
- 2 MEDIUM at a time

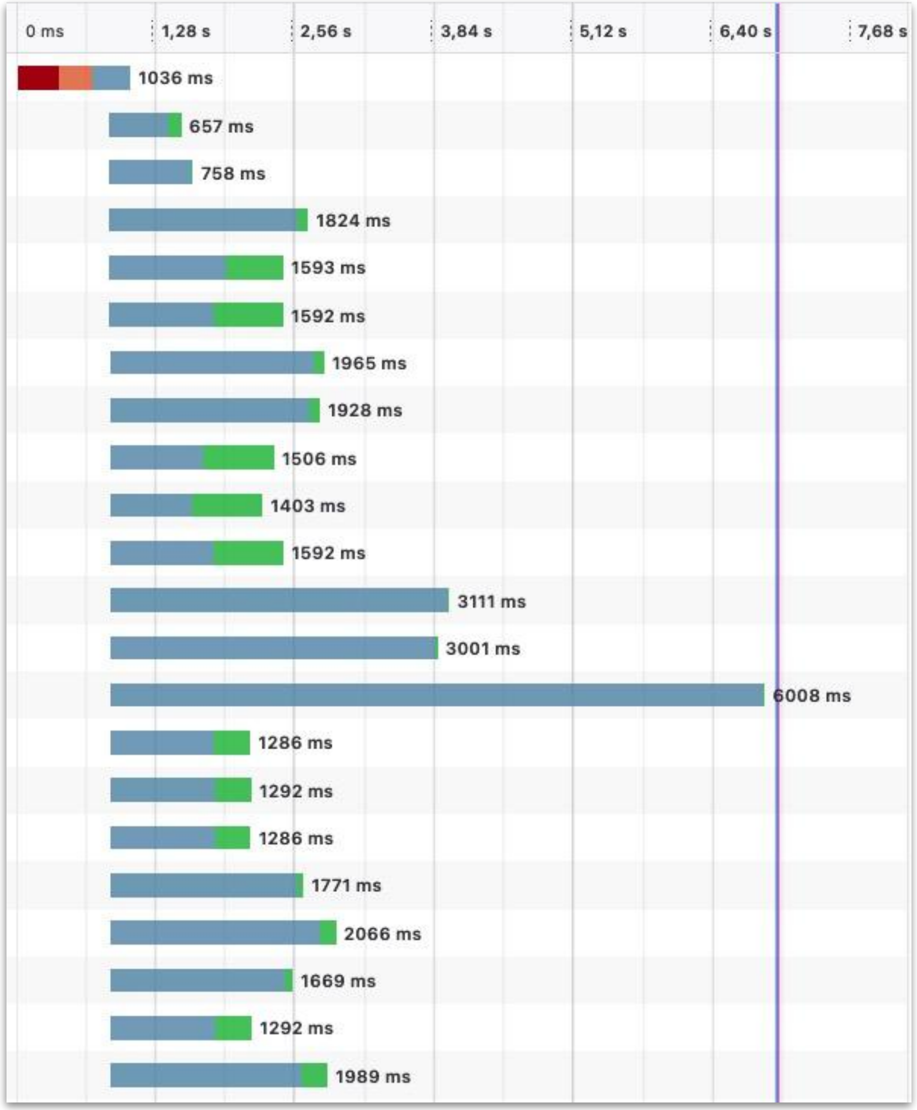
While blocking JS **or CSS** ~*anywhere* is busy

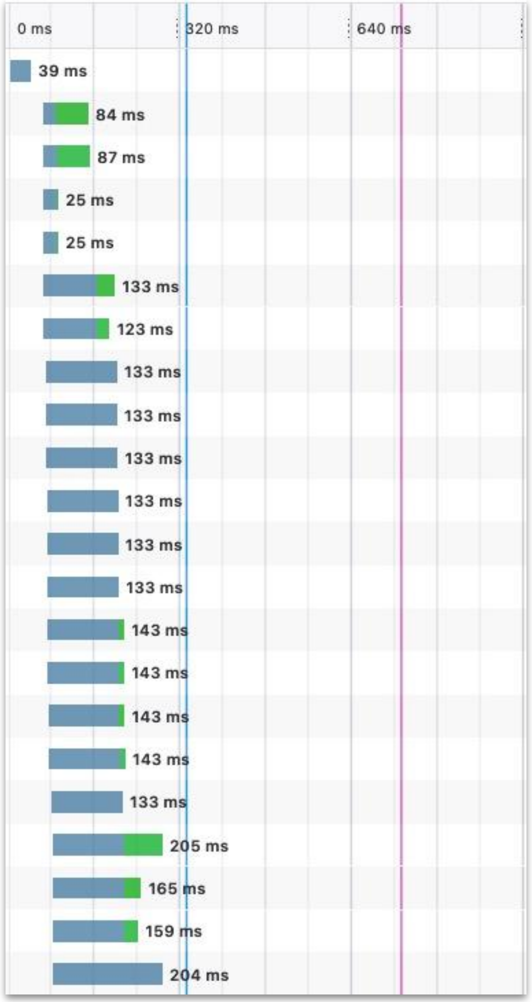
- Only MEDIUM/LOW/LOWEST if fewer than 2 things in flight
    - With the exception of async/defer JS, those always get requested asap

Firefox doesn’t do Tight Mode in HTTP/2 and /3



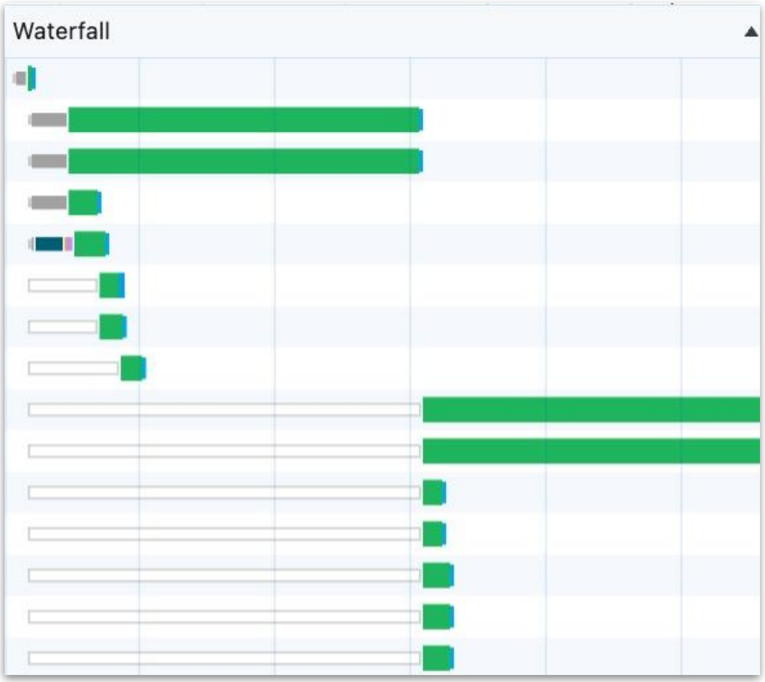


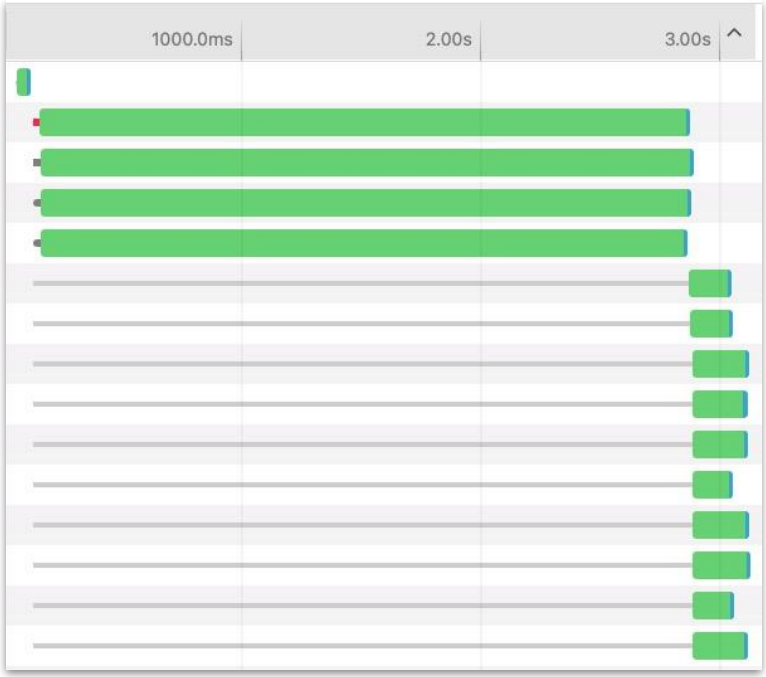


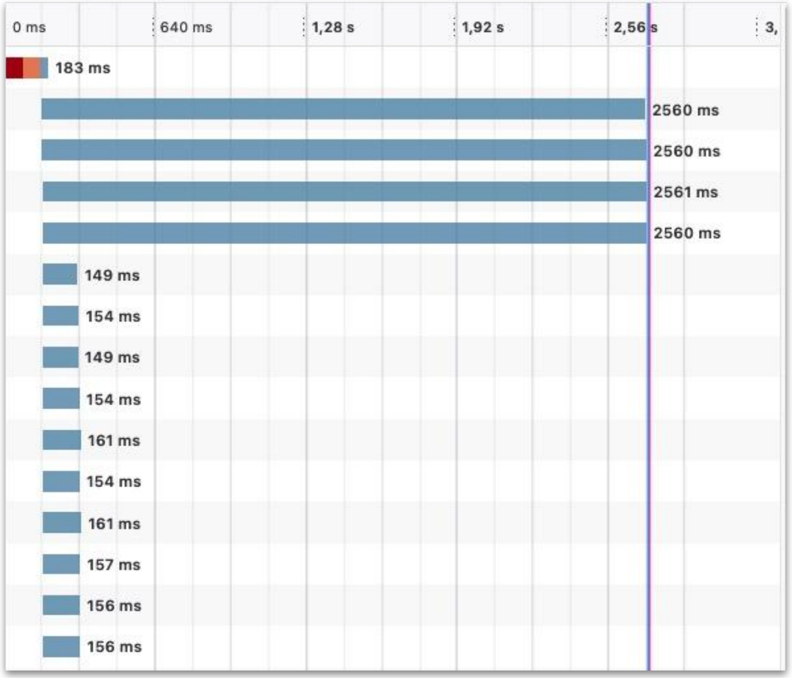


Exact same HTML,

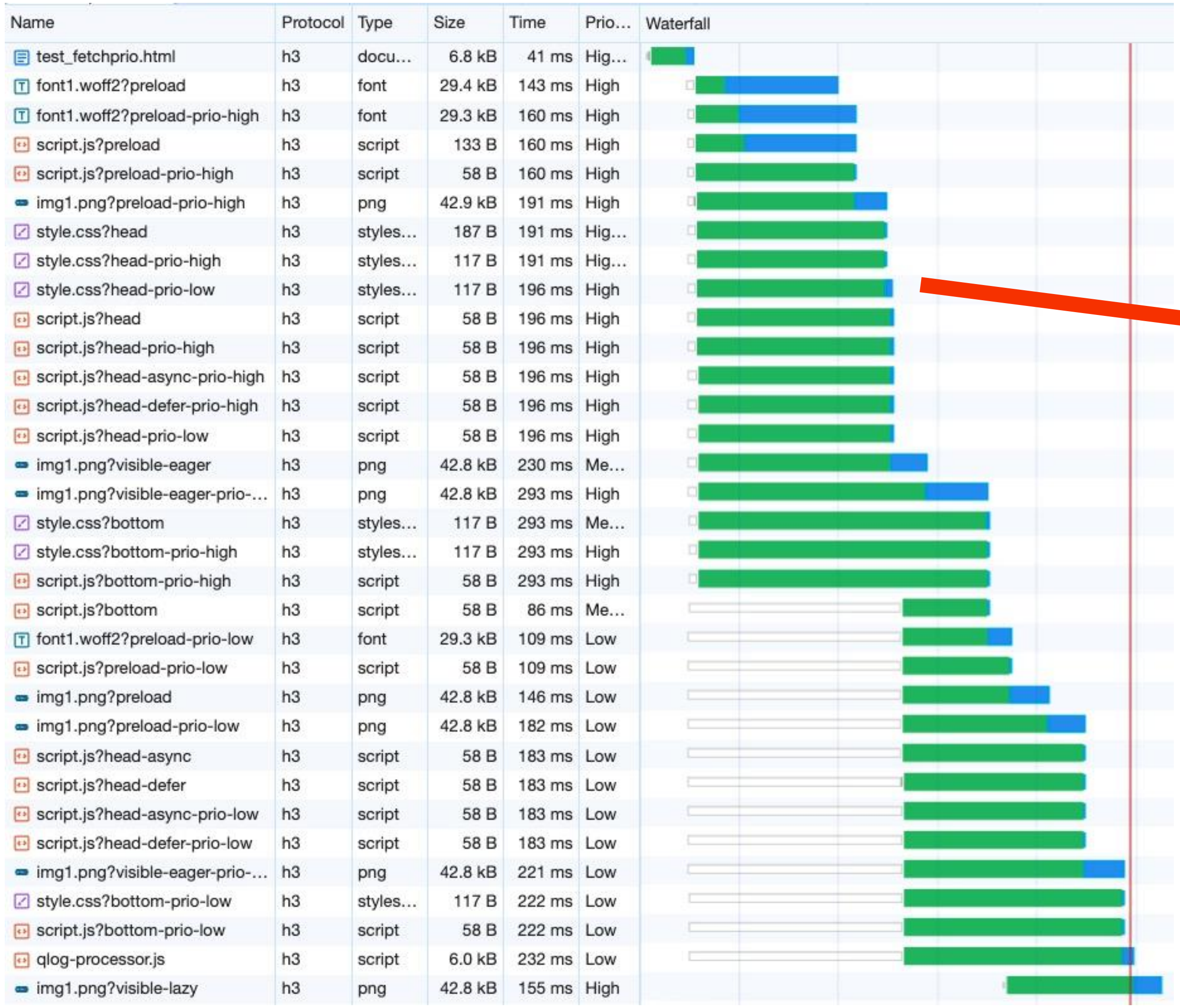
radically different behaviour







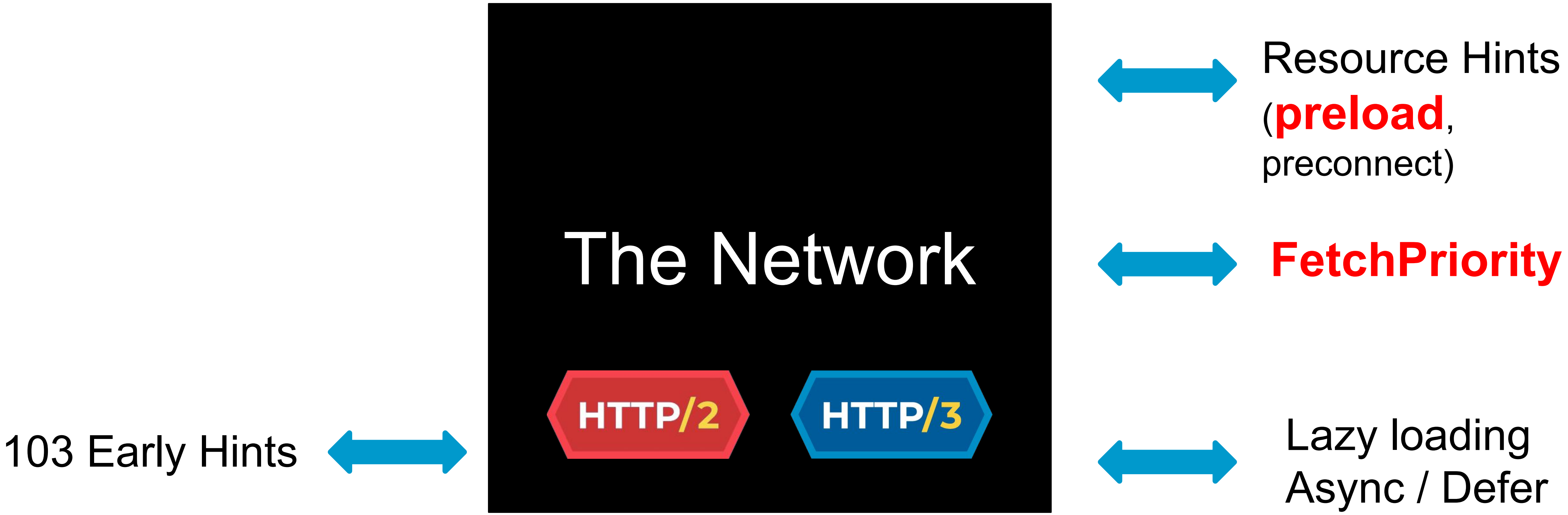
How to fix wrong browser behaviour?



Name	Protocol	Type	Size	Time	Prio…	Waterfall
test_fetchprio.html	h3	docu…	6.8 kB	41 ms	Hig…	
font1.woff2?preload	h3	font	29.4 kB	143 ms	High	
font1.woff2?preload-prio-high	h3	font	29.3 kB	160 ms	High	
script.js?preload	h3	script	133 B	160 ms	High	
script.js?preload-prio-high	h3	script	58 B	160 ms	High	
img1.png?preload-prio-high	h3	png	42.9 kB	191 ms	High	
style.css?head	h3	styles…	187 B	191 ms	Hig…	
style.css?head-prio-high	h3	styles…	117 B	191 ms	Hig…	
style.css?head-prio-low	h3	styles…	117 B	196 ms	High	
script.js?head	h3	script	58 B	196 ms	High	
script.js?head-prio-high	h3	script	58 B	196 ms	High	
script.js?head-async-prio-high	h3	script	58 B	196 ms	High	
script.js?head-defer-prio-high	h3	script	58 B	196 ms	High	
script.js?head-prio-low	h3	script	58 B	196 ms	High	
img1.png?visible-eager	h3	png	42.8 kB	230 ms	Me…	
img1.png?visible-eager-prio-…	h3	png	42.8 kB	293 ms	High	
style.css?bottom	h3	styles…	117 B	293 ms	Me…	
style.css?bottom-prio-high	h3	styles…	117 B	293 ms	High	
script.js?bottom-prio-high	h3	script	58 B	293 ms	High	
script.js?bottom	h3	script	58 B	86 ms	Me…	
font1.woff2?preload-prio-low	h3	font	29.3 kB	109 ms	Low	
script.js?preload-prio-low	h3	script	58 B	109 ms	Low	
img1.png?preload	h3	png	42.8 kB	146 ms	Low	
img1.png?preload-prio-low	h3	png	42.8 kB	182 ms	Low	
script.js?head-async	h3	script	58 B	183 ms	Low	
script.js?head-defer	h3	script	58 B	183 ms	Low	
script.js?head-async-prio-low	h3	script	58 B	183 ms	Low	
script.js?head-defer-prio-low	h3	script	58 B	183 ms	Low	
img1.png?visible-eager-prio-…	h3	png	42.8 kB	221 ms	Low	
style.css?bottom-prio-low	h3	styles…	117 B	222 ms	Low	
script.js?bottom-prio-low	h3	script	58 B	222 ms	Low	
qlog-processor.js	h3	script	6.0 kB	232 ms	Low	
img1.png?visible-lazy	h3	png	42.8 kB	155 ms	High	

This should actually be down there

(or vice versa)



The Network

HTTP/2

HTTP/3

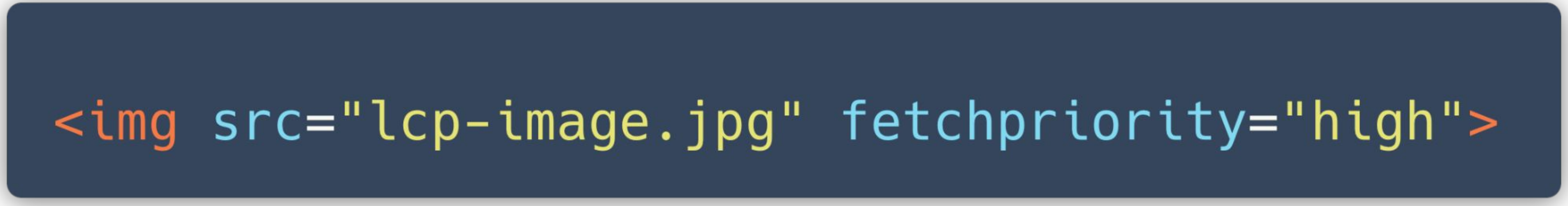
103 Early Hints

Resource Hints (**preload**, preconnect)

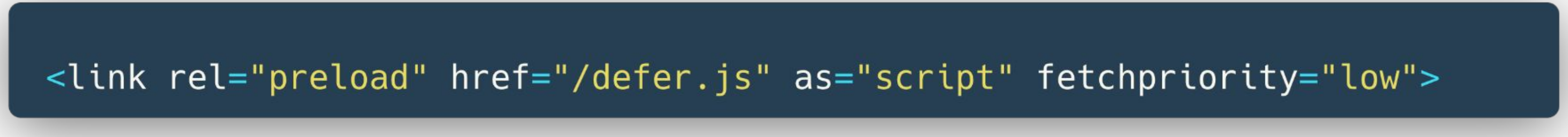
FetchPriority

Lazy loading
Async / Defer

FetchPriority to the rescue!?



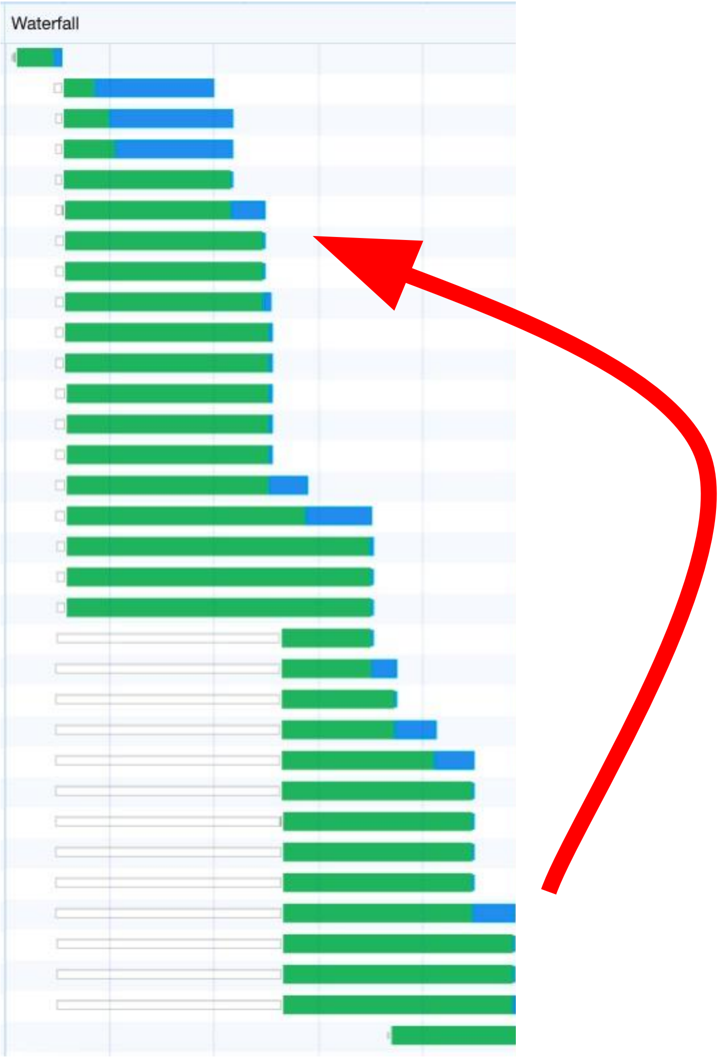
```html
<img src="lcp-image.jpg" fetchpriority="high">
```



```html
<link rel="preload" href="/defer.js" as="script" fetchpriority="low">
```

https://web.dev/fetch-priority

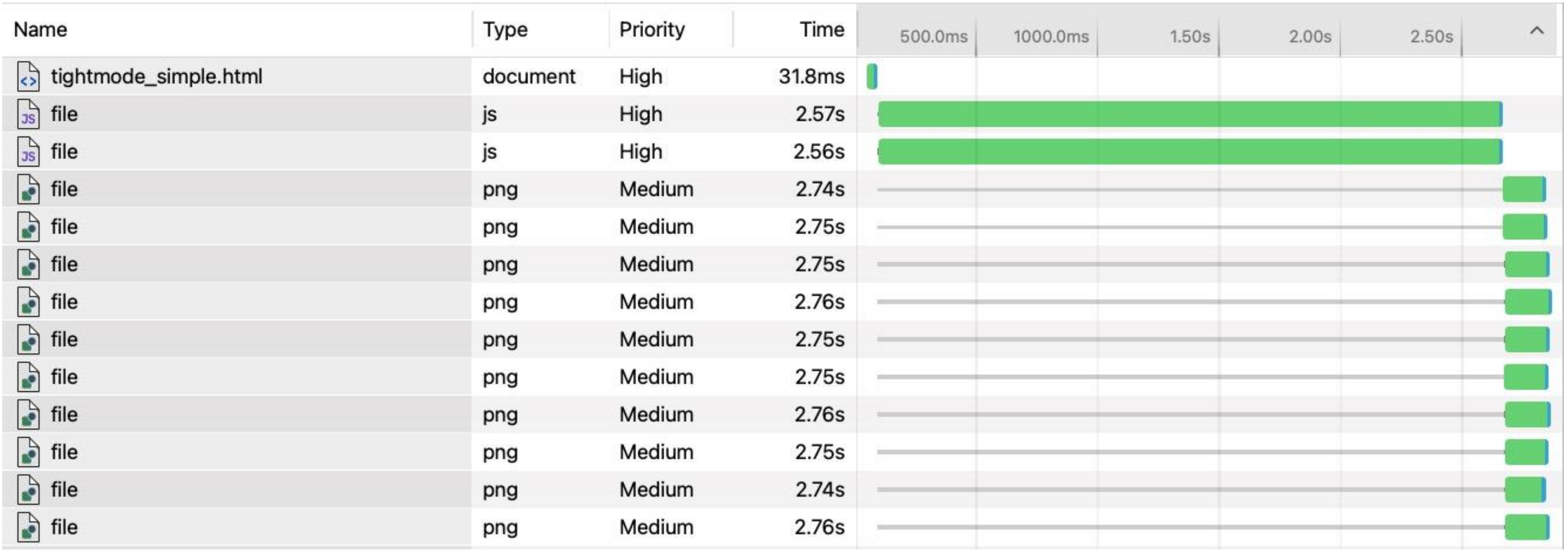
How to get stuff INTO tight mode?



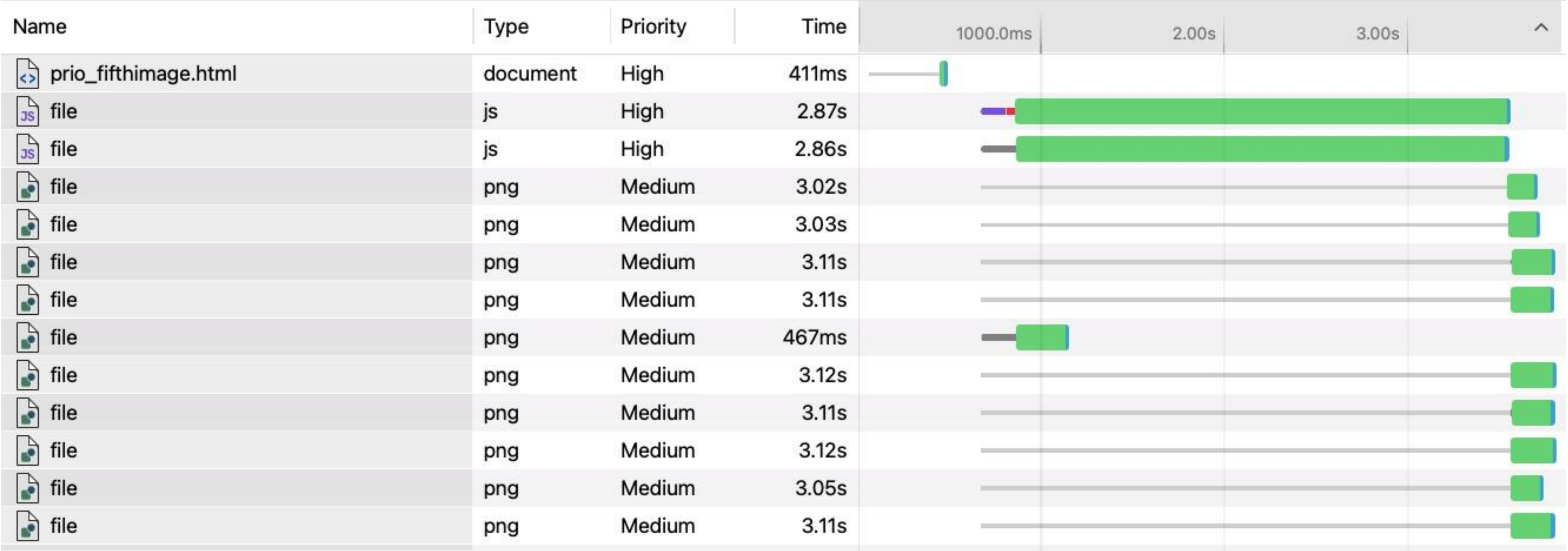
fetchpriority=high

- Images
- Defer/Async JS
- JS on the bottom of the <body>

- Images



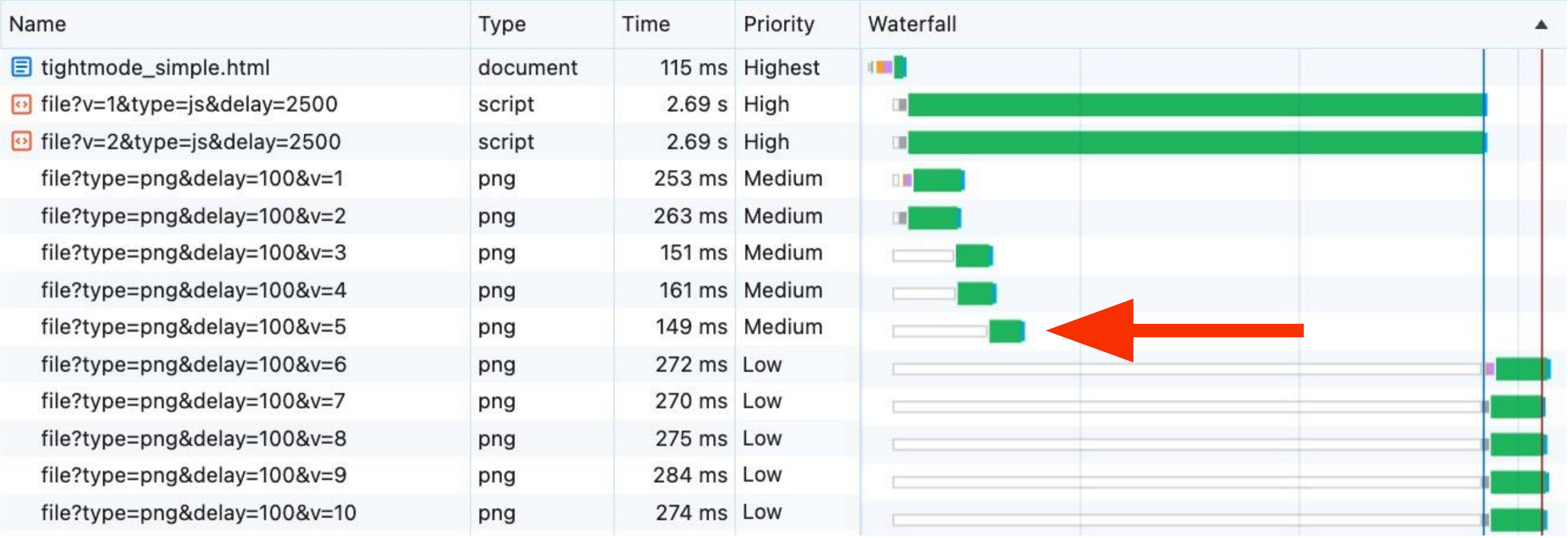
Name	Type	Priority	Time
tightmode_simple.html	document	High	31.8ms
file	js	High	2.57s
file	js	High	2.56s
file	png	Medium	2.74s
file	png	Medium	2.75s
file	png	Medium	2.75s
file	png	Medium	2.76s
file	png	Medium	2.75s
file	png	Medium	2.75s
file	png	Medium	2.76s
file	png	Medium	2.75s
file	png	Medium	2.74s
file	png	Medium	2.76s



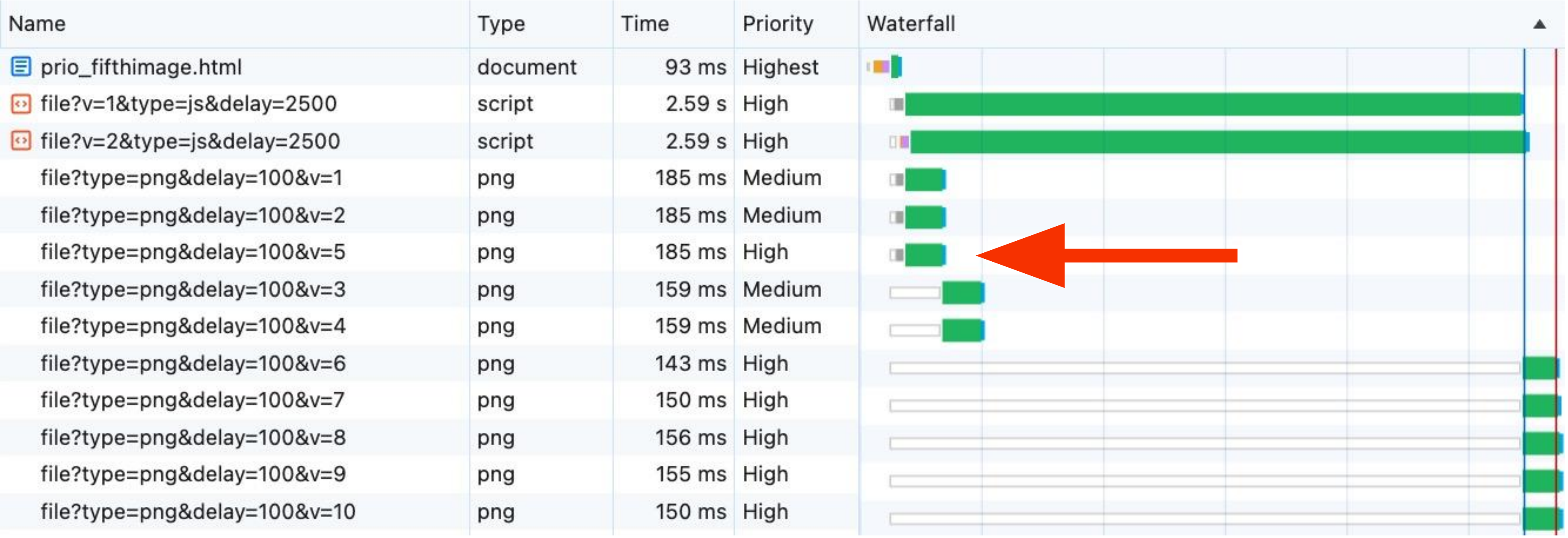
Name	Type	Priority	Time
prio_fifthimage.html	document	High	411ms
file	js	High	2.87s
file	js	High	2.86s
file	png	Medium	3.02s
file	png	Medium	3.03s
file	png	Medium	3.11s
file	png	Medium	3.11s
file	png	Medium	467ms
file	png	Medium	3.12s
file	png	Medium	3.11s
file	png	Medium	3.12s
file	png	Medium	3.05s
file	png	Medium	3.11s

fetchpriority= high

causes image 5 to load in tight mode



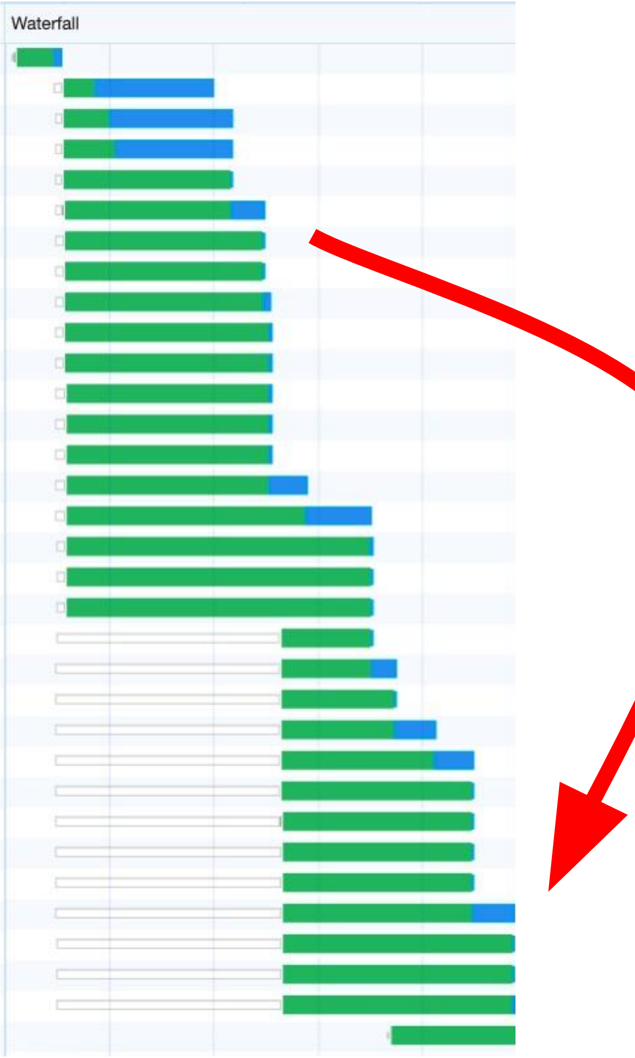
Name	Type	Time	Priority	Waterfall
tightmode_simple.html	document	115 ms	Highest	
file?v=1&type=js&delay=2500	script	2.69 s	High	
file?v=2&type=js&delay=2500	script	2.69 s	High	
file?type=png&delay=100&v=1	png	253 ms	Medium	
file?type=png&delay=100&v=2	png	263 ms	Medium	
file?type=png&delay=100&v=3	png	151 ms	Medium	
file?type=png&delay=100&v=4	png	161 ms	Medium	
file?type=png&delay=100&v=5	png	149 ms	Medium	
file?type=png&delay=100&v=6	png	272 ms	Low	
file?type=png&delay=100&v=7	png	270 ms	Low	
file?type=png&delay=100&v=8	png	275 ms	Low	
file?type=png&delay=100&v=9	png	284 ms	Low	
file?type=png&delay=100&v=10	png	274 ms	Low	



Name	Type	Time	Priority	Waterfall
prio_fifthimage.html	document	93 ms	Highest	
file?v=1&type=js&delay=2500	script	2.59 s	High	
file?v=2&type=js&delay=2500	script	2.59 s	High	
file?type=png&delay=100&v=1	png	185 ms	Medium	
file?type=png&delay=100&v=2	png	185 ms	Medium	
file?type=png&delay=100&v=5	png	185 ms	High	
file?type=png&delay=100&v=3	png	159 ms	Medium	
file?type=png&delay=100&v=4	png	159 ms	Medium	
file?type=png&delay=100&v=6	png	143 ms	High	
file?type=png&delay=100&v=7	png	150 ms	High	
file?type=png&delay=100&v=8	png	156 ms	High	
file?type=png&delay=100&v=9	png	155 ms	High	
file?type=png&delay=100&v=10	png	150 ms	High	

image 5 is requested before 3 and 4

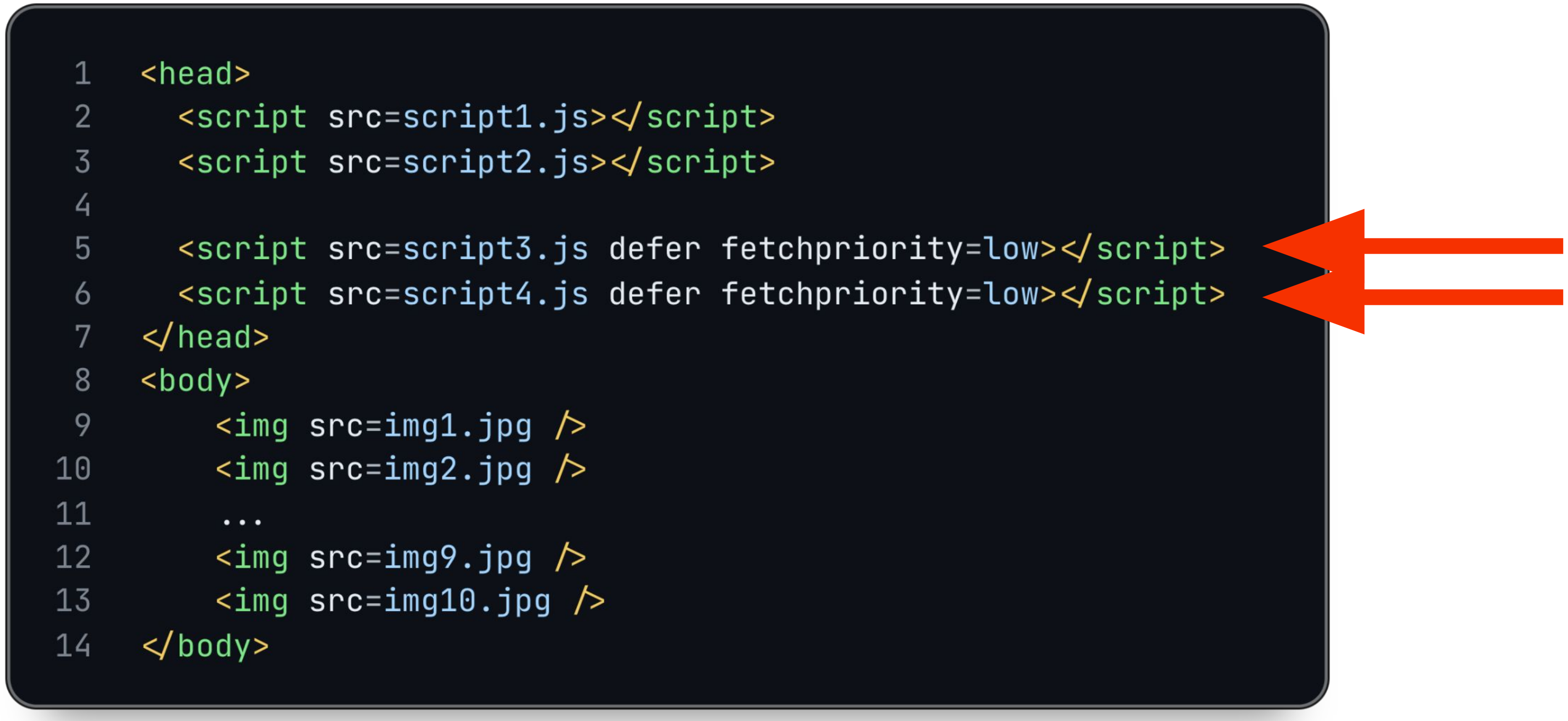
How to get stuff OUT OF tight mode?



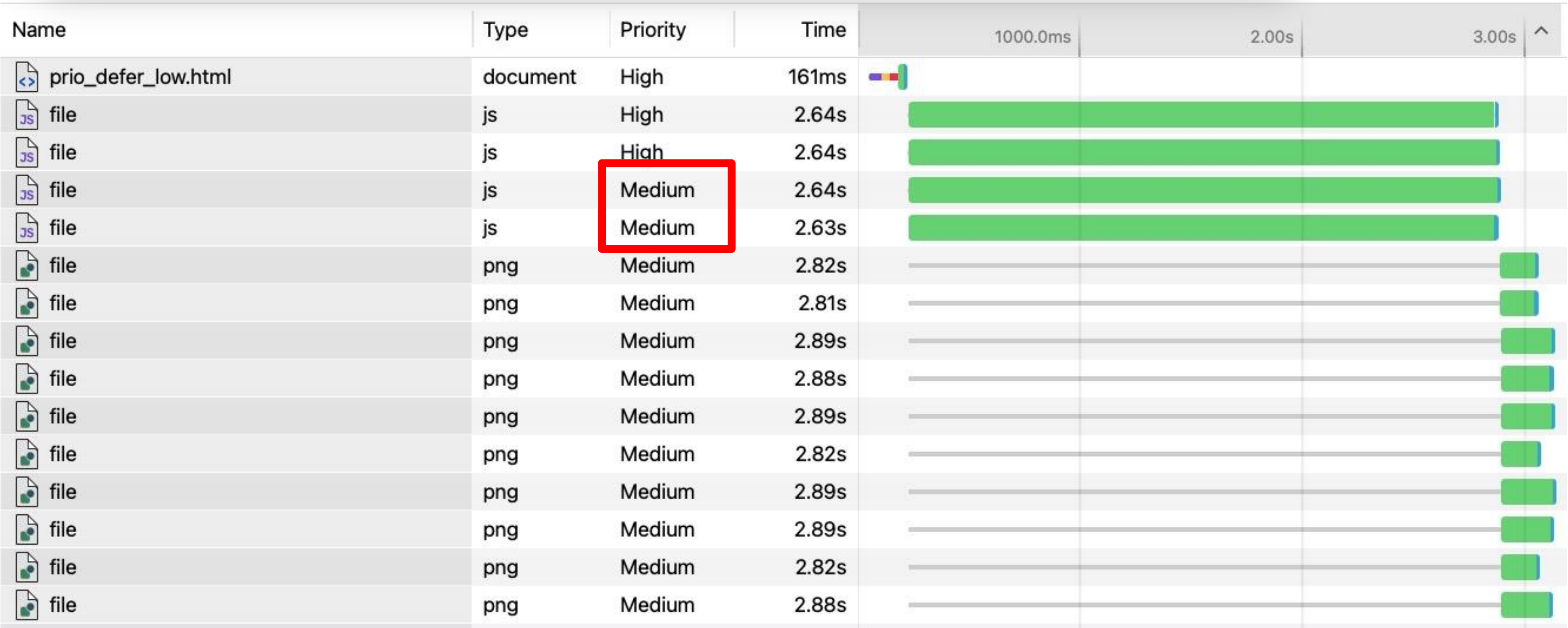
fetchpriority=low

- First 5 images
- JS early and CSS late in <body>
- Preloaded fonts
- Preloaded async/defer JS

- **NOTHING AT ALL?!?**

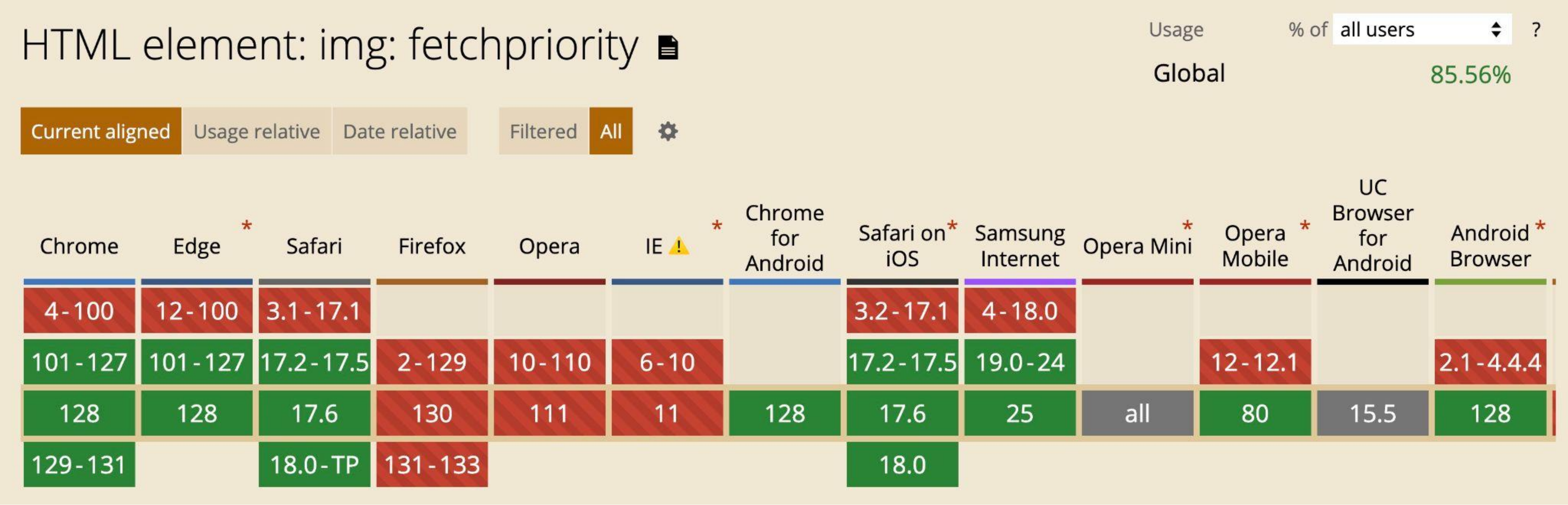


```
<head>
  <script src=script1.js></script>
  <script src=script2.js></script>

  <script src=script3.js defer fetchpriority=low></script>
  <script src=script4.js defer fetchpriority=low></script>
</head>
<body>
    <img src=img1.jpg />
    <img src=img2.jpg />
    ...
    <img src=img9.jpg />
    <img src=img10.jpg />
</body>
```



Name	Type	Priority	Time
prio_defer_low.html	document	High	161ms
file	js	High	2.64s
file	js	High	2.64s
file	js	Medium	2.64s
file	js	Medium	2.63s
file	png	Medium	2.82s
file	png	Medium	2.81s
file	png	Medium	2.89s
file	png	Medium	2.88s
file	png	Medium	2.89s
file	png	Medium	2.82s
file	png	Medium	2.89s
file	png	Medium	2.89s
file	png	Medium	2.82s
file	png	Medium	2.88s

Oh the Irony



HTML element: img: fetchpriority

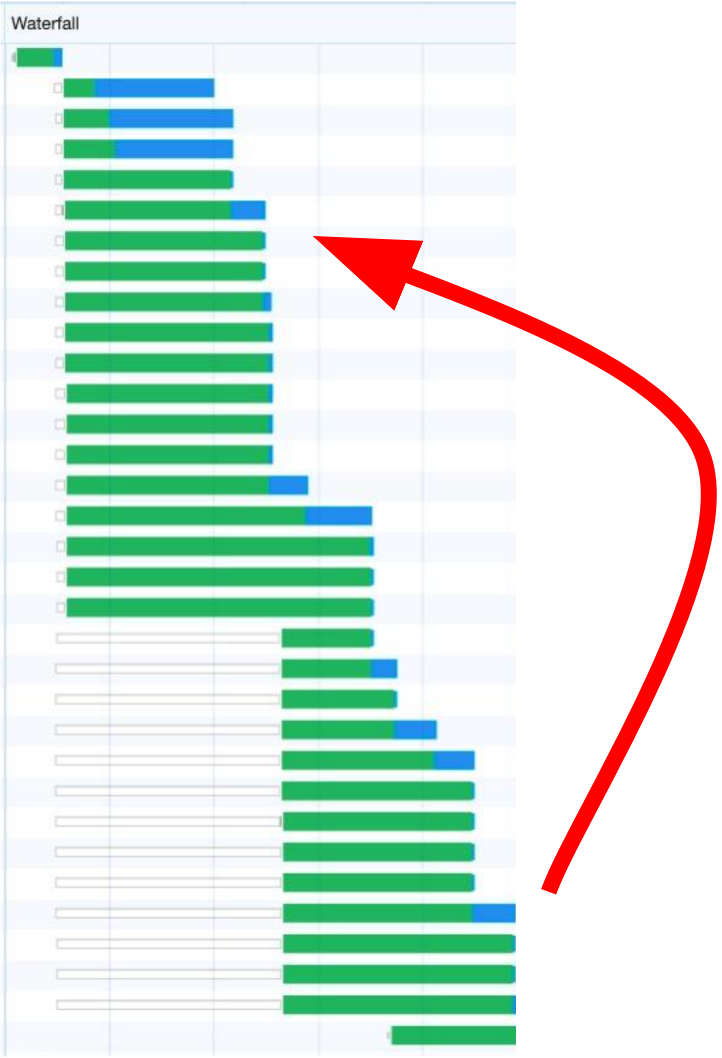
Usage % of all users

Global 85.56%

Chrome	Edge *	Safari	Firefox	Opera	IE ⚠ *	Chrome for Android	Safari on iOS *	Samsung Internet	Opera Mini *	Opera Mobile *	UC Browser for Android	Android Browser *
4-100	12-100	3.1-17.1					3.2-17.1	4-18.0				
101-127	101-127	17.2-17.5	2-129	10-110	6-10		17.2-17.5	19.0-24		12-12.1		2.1-4.4.4
128	128	17.6	130	111	11	128	17.6	25	all	80	15.5	128
129-131		18.0-TP	131-133				18.0					

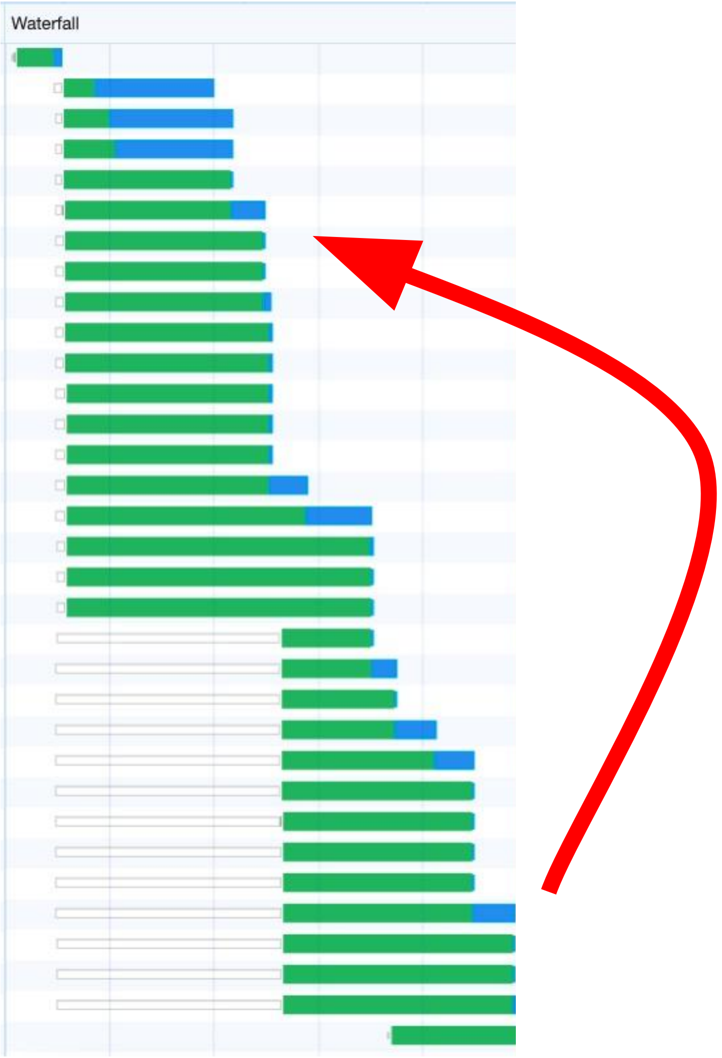
https://bugzilla.mozilla.org/show_bug.cgi?id=1797715

How to get stuff INTO tight mode?

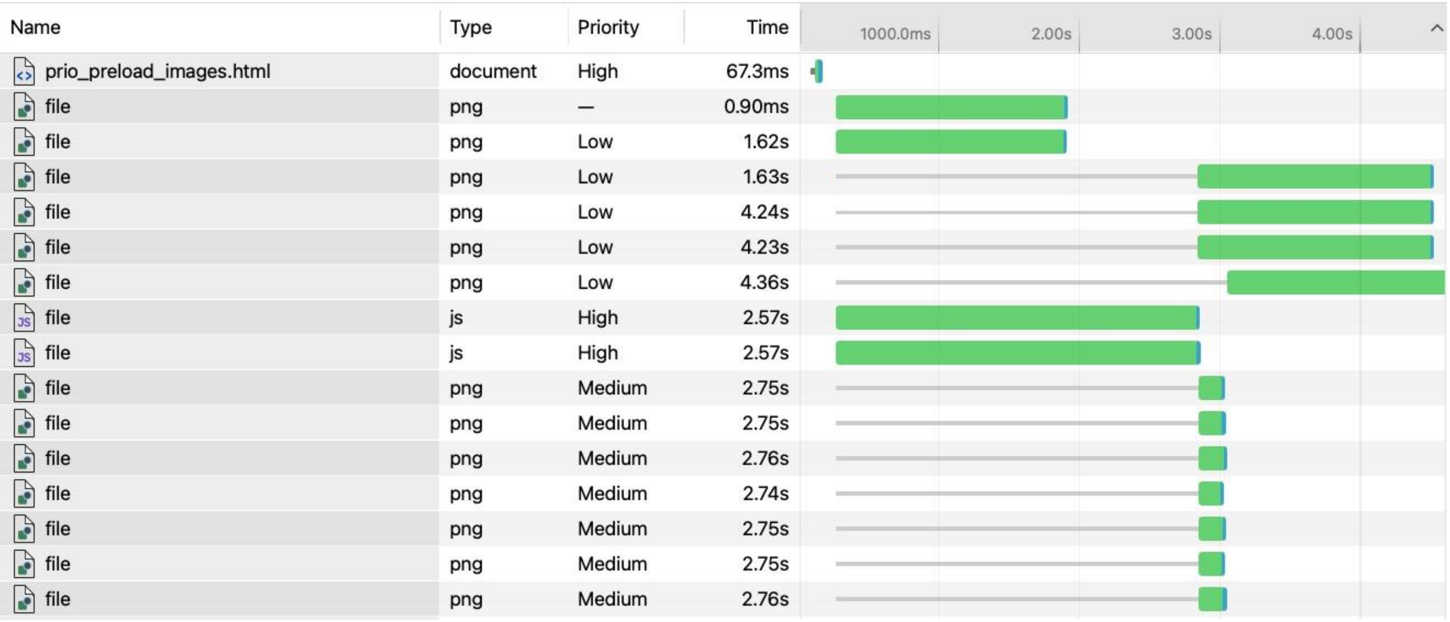


<link rel=**preload** src=lcp.jpg as=image />

How to get stuff INTO tight mode?



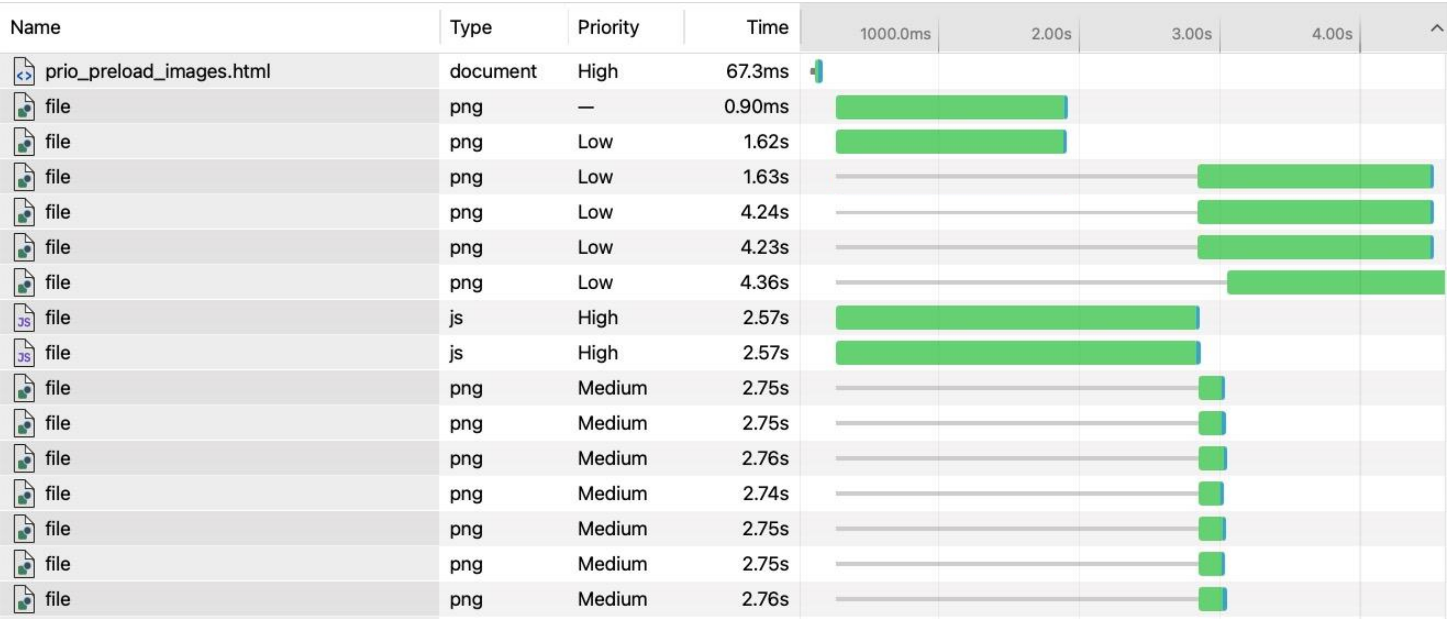
```
<link rel=preload src=lcp.jpg as=image />
```



Name	Type	Priority	Time
prio_preload_images.html	document	High	67.3ms
file	png	—	0.90ms
file	png	Low	1.62s
file	png	Low	1.63s
file	png	Low	4.24s
file	png	Low	4.23s
file	png	Low	4.36s
file	js	High	2.57s
file	js	High	2.57s
file	png	Medium	2.75s
file	png	Medium	2.75s
file	png	Medium	2.76s
file	png	Medium	2.74s
file	png	Medium	2.75s
file	png	Medium	2.75s
file	png	Medium	2.76s

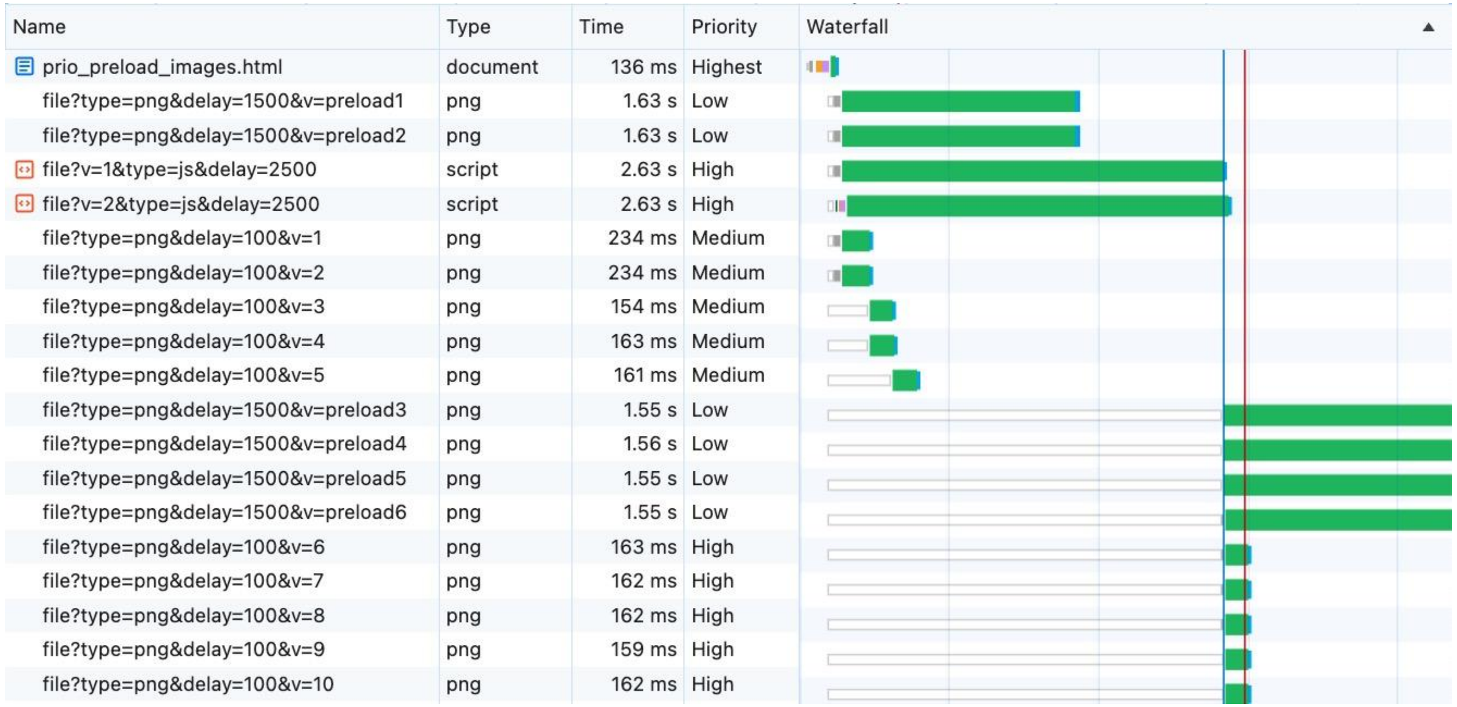
Preload 6 images on top

Only 2 preloads fire at the start, other 4 don’t



Name	Type	Priority	Time
prio_preload_images.html	document	High	67.3ms
file	png	—	0.90ms
file	png	Low	1.62s
file	png	Low	1.63s
file	png	Low	4.24s
file	png	Low	4.23s
file	png	Low	4.36s
file	js	High	2.57s
file	js	High	2.57s
file	png	Medium	2.75s
file	png	Medium	2.75s
file	png	Medium	2.76s
file	png	Medium	2.74s
file	png	Medium	2.75s
file	png	Medium	2.75s
file	png	Medium	2.76s

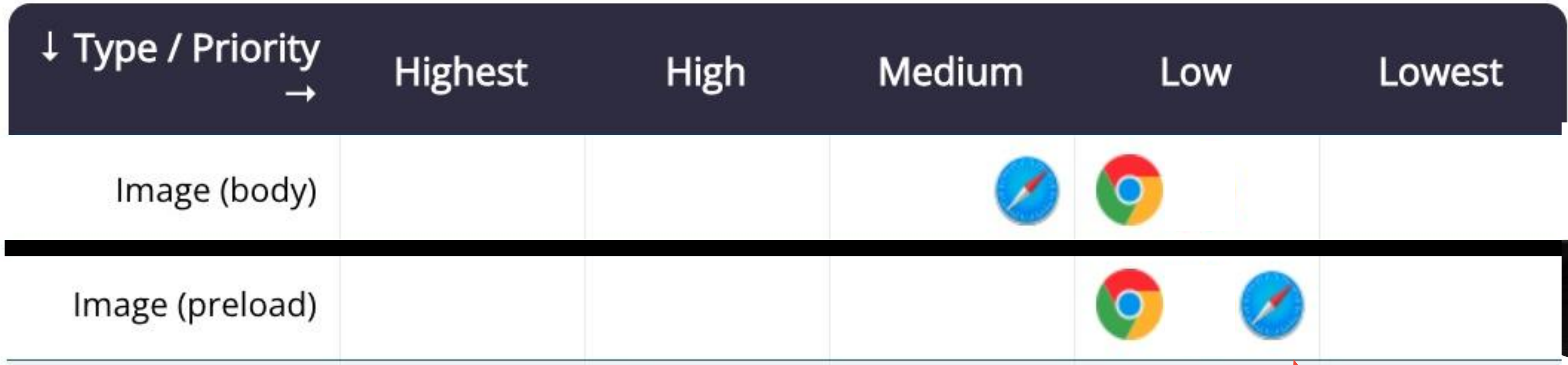
Preload 6 images on top



Name	Type	Time	Priority	Waterfall
prio_preload_images.html	document	136 ms	Highest	
file?type=png&delay=1500&v=preload1	png	1.63 s	Low	
file?type=png&delay=1500&v=preload2	png	1.63 s	Low	
file?v=1&type=js&delay=2500	script	2.63 s	High	
file?v=2&type=js&delay=2500	script	2.63 s	High	
file?type=png&delay=100&v=1	png	234 ms	Medium	
file?type=png&delay=100&v=2	png	234 ms	Medium	
file?type=png&delay=100&v=3	png	154 ms	Medium	
file?type=png&delay=100&v=4	png	163 ms	Medium	
file?type=png&delay=100&v=5	png	161 ms	Medium	
file?type=png&delay=1500&v=preload3	png	1.55 s	Low	
file?type=png&delay=1500&v=preload4	png	1.56 s	Low	
file?type=png&delay=1500&v=preload5	png	1.55 s	Low	
file?type=png&delay=1500&v=preload6	png	1.55 s	Low	
file?type=png&delay=100&v=6	png	163 ms	High	
file?type=png&delay=100&v=7	png	162 ms	High	
file?type=png&delay=100&v=8	png	162 ms	High	
file?type=png&delay=100&v=9	png	159 ms	High	
file?type=png&delay=100&v=10	png	162 ms	High	

Basic tight mode “2 low prio in flight at the same time” logic

Preload doesn't *increase* priority by itself

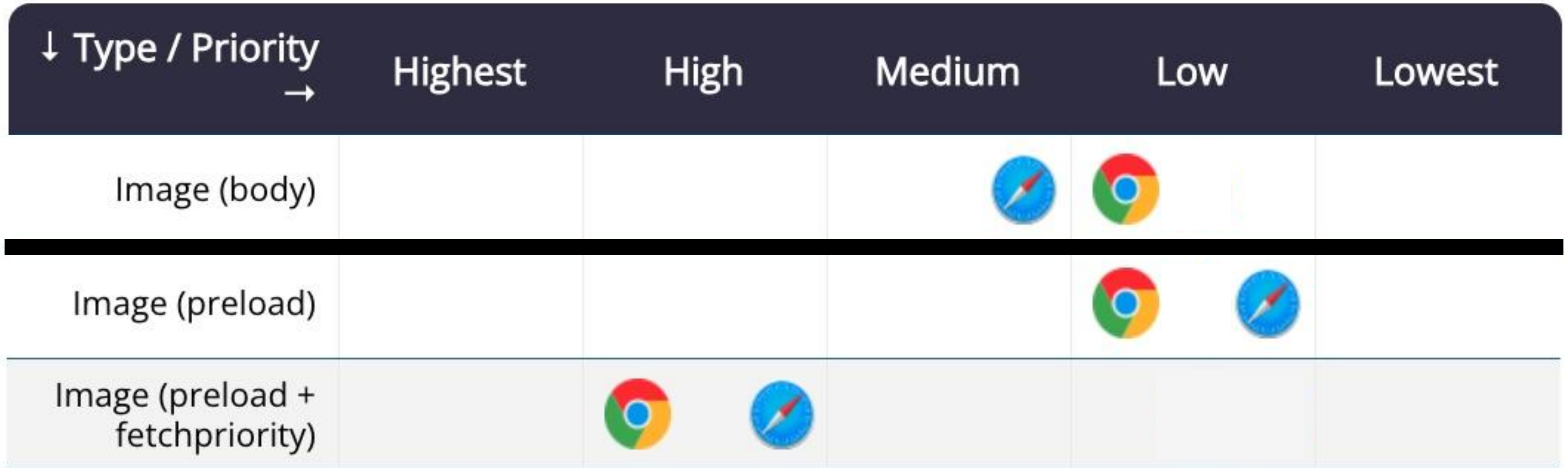


↓ Type / Priority →	Highest	High	Medium	Low	Lowest
Image (body)			🧭	🌐	
Image (preload)				🌐 🧭	

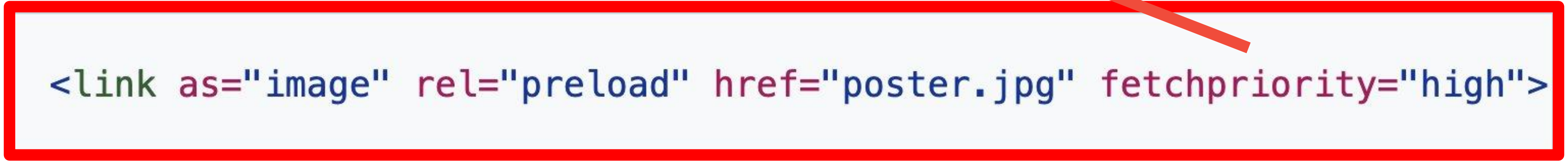
Image preload actually
LOWERS
priority in Safari

https://calendar.perfplanet.com/2022/http-3-prioritization-demystified/
https://bugs.chromium.org/p/chromium/issues/detail?id=1431169

You need fetchpriority=high for that

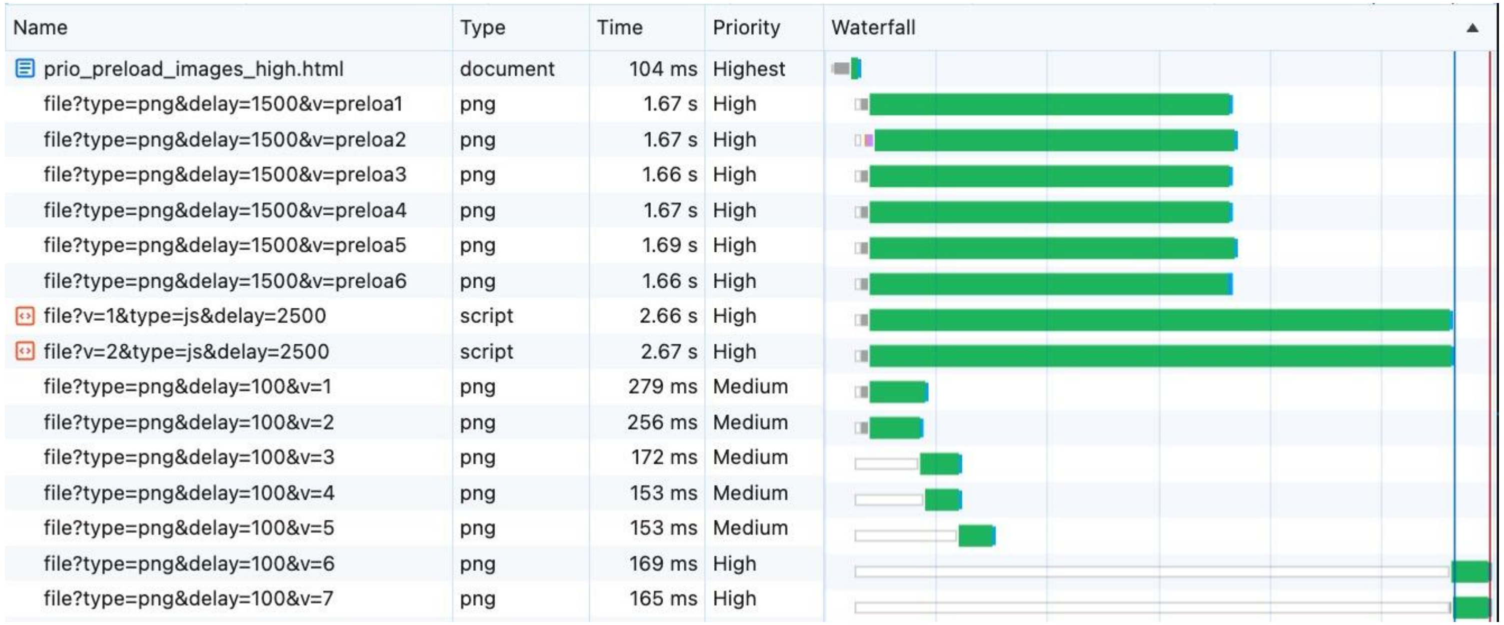


↓ Type / Priority →	Highest	High	Medium	Low	Lowest
Image (body)			Safari	Chrome	
Image (preload)				Chrome Safari	
Image (preload + fetchpriority)		Chrome Safari			

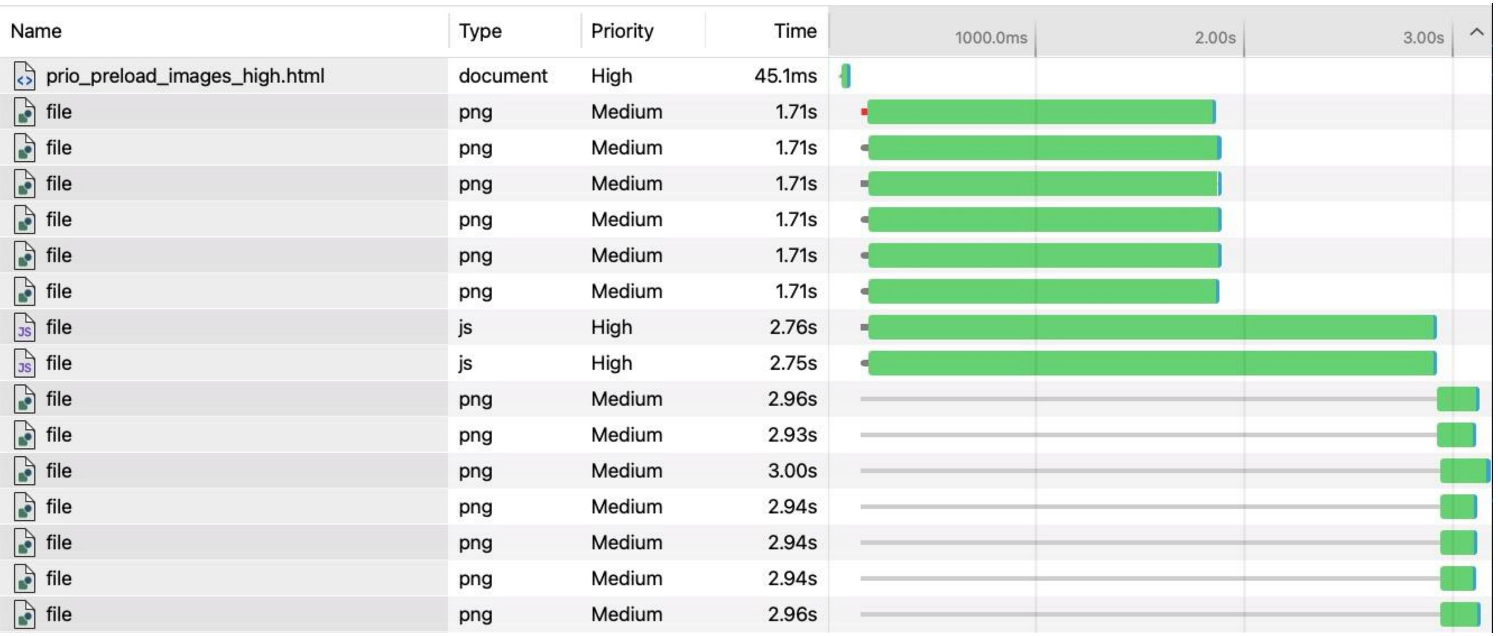


```
<link as="image" rel="preload" href="poster.jpg" fetchpriority="high">
```

https://calendar.perfplanet.com/2022/http-3-prioritization-demystified/
https://bugs.chromium.org/p/chromium/issues/detail?id=1431169



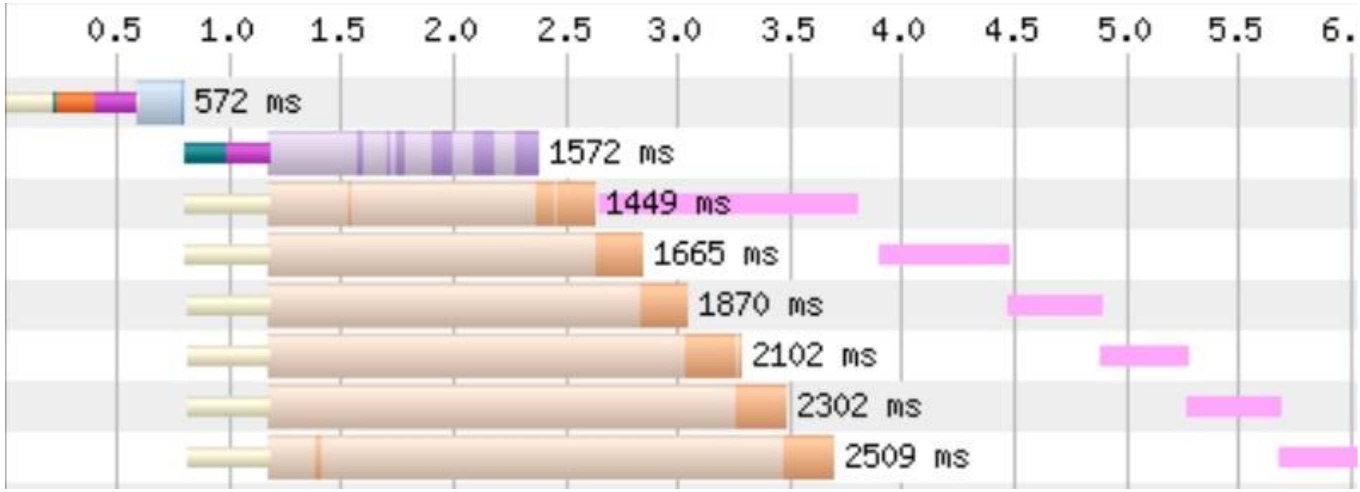
Name	Type	Time	Priority	Waterfall
prio_preload_images_high.html	document	104 ms	Highest	
file?type=png&delay=1500&v=preloa1	png	1.67 s	High	
file?type=png&delay=1500&v=preloa2	png	1.67 s	High	
file?type=png&delay=1500&v=preloa3	png	1.66 s	High	
file?type=png&delay=1500&v=preloa4	png	1.67 s	High	
file?type=png&delay=1500&v=preloa5	png	1.69 s	High	
file?type=png&delay=1500&v=preloa6	png	1.66 s	High	
file?v=1&type=js&delay=2500	script	2.66 s	High	
file?v=2&type=js&delay=2500	script	2.67 s	High	
file?type=png&delay=100&v=1	png	279 ms	Medium	
file?type=png&delay=100&v=2	png	256 ms	Medium	
file?type=png&delay=100&v=3	png	172 ms	Medium	
file?type=png&delay=100&v=4	png	153 ms	Medium	
file?type=png&delay=100&v=5	png	153 ms	Medium	
file?type=png&delay=100&v=6	png	169 ms	High	
file?type=png&delay=100&v=7	png	165 ms	High	



Name	Type	Priority	Time	1000.0ms 2.00s 3.00s
prio_preload_images_high.html	document	High	45.1ms	
file	png	Medium	1.71s	
file	png	Medium	1.71s	
file	png	Medium	1.71s	
file	png	Medium	1.71s	
file	png	Medium	1.71s	
file	png	Medium	1.71s	
file	js	High	2.76s	
file	js	High	2.75s	
file	png	Medium	2.96s	
file	png	Medium	2.93s	
file	png	Medium	3.00s	
file	png	Medium	2.94s	
file	png	Medium	2.94s	
file	png	Medium	2.94s	
file	png	Medium	2.96s	

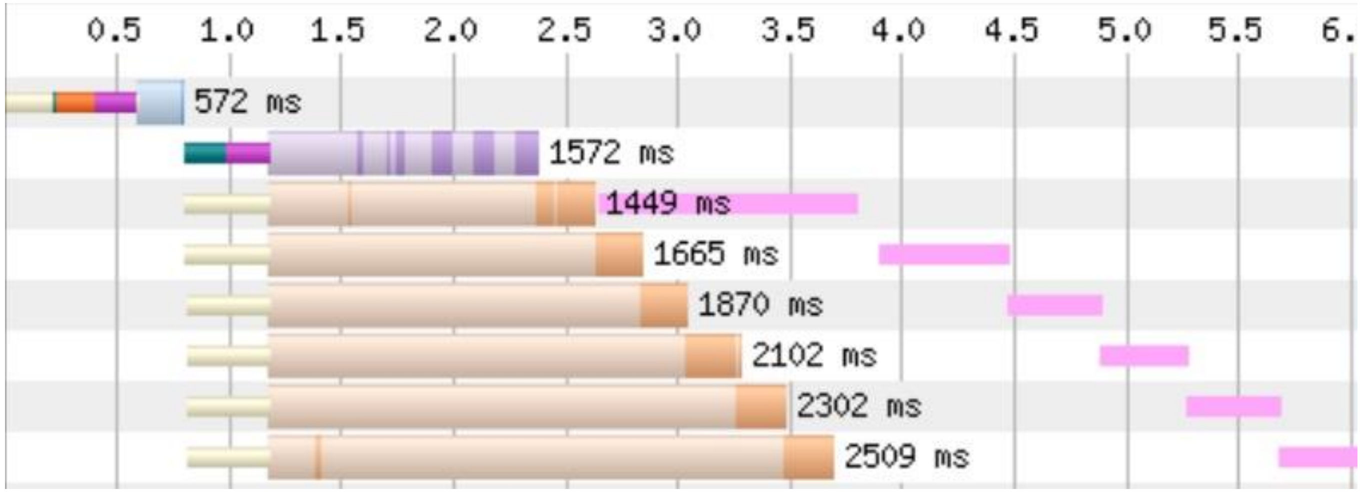
Preload 6 images with fetchpriority =high

Now they are all requested during tight mode



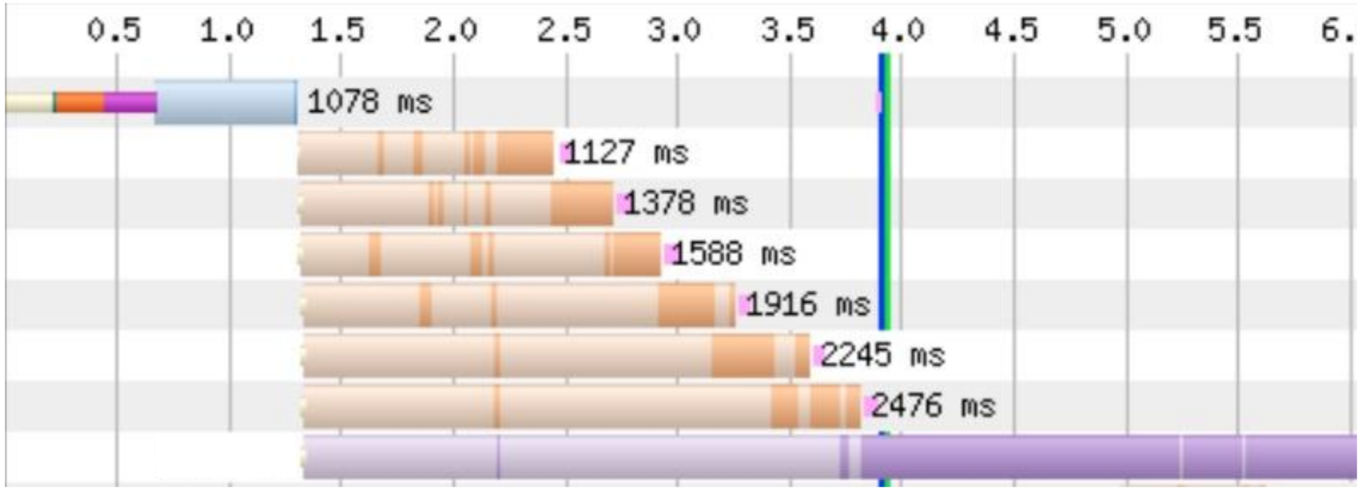
preload on **top** of <head> +
fetchpriority = high

= loaded **before** parser-blocking JS



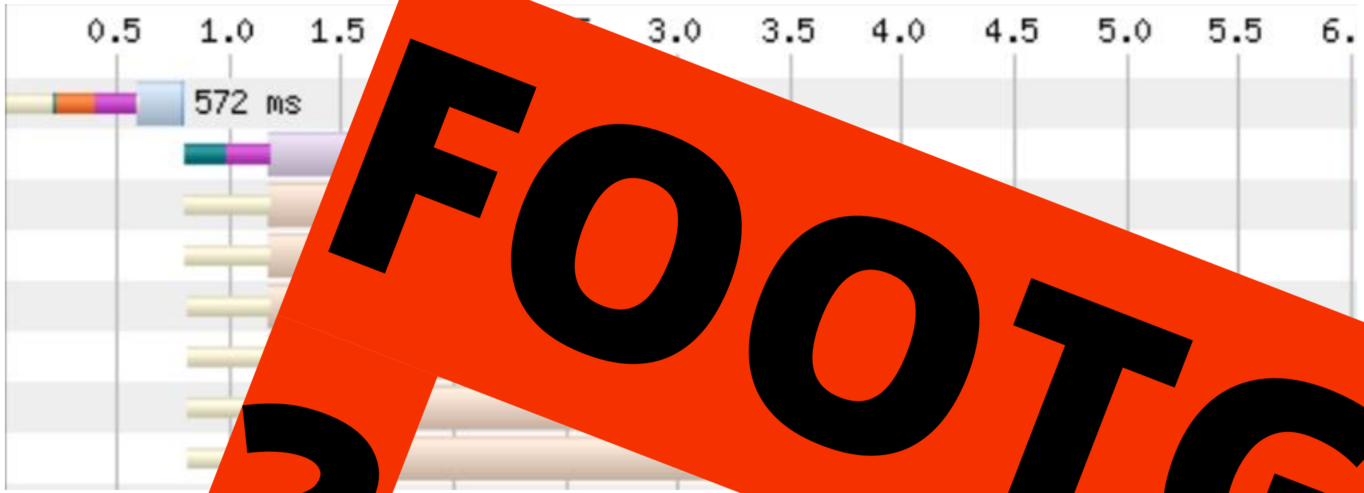
preload on **top** of <head> +
fetchpriority = high

= loaded **before** parser-blocking JS



preload on **bottom** of <head> +
fetchpriority = high

= loaded **after** parser-blocking JS

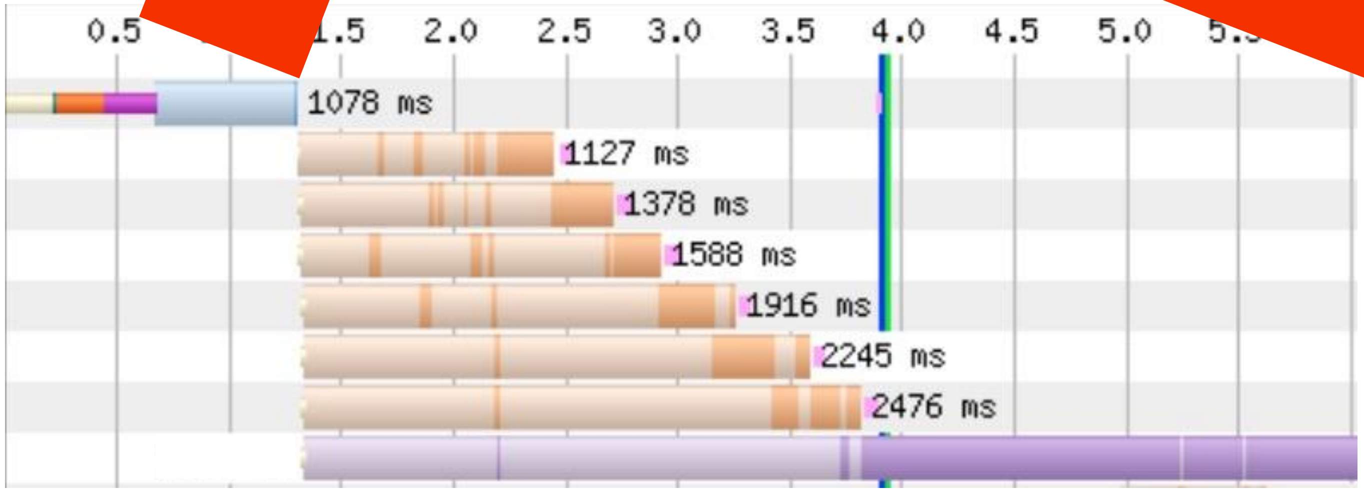


preload on **top** of <head> + fetchpriority = high

ded **before** parser-blocking JS

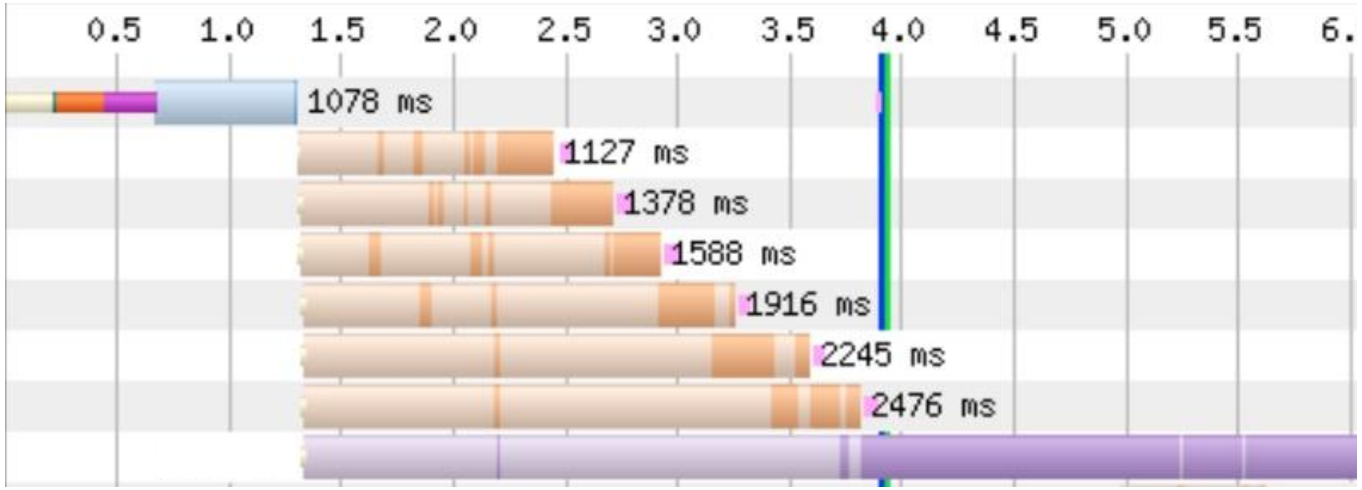
FOOTGUN!

?



preloa of <head> + fetchpriority

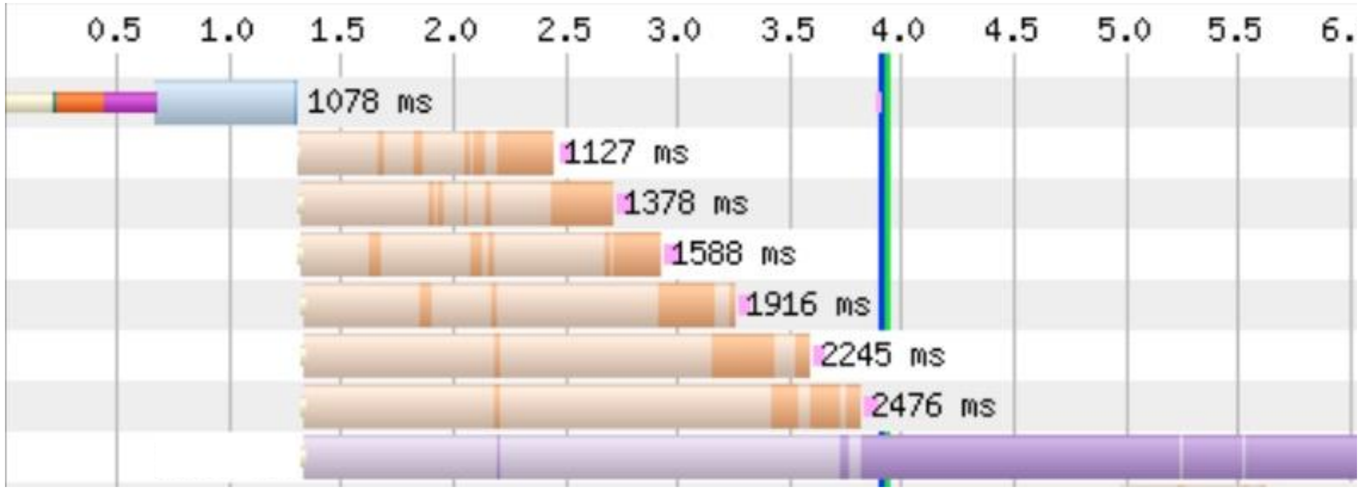
= loaded **after** parser-blocking JS



preload on **bottom** of <head> + fetchpriority = high

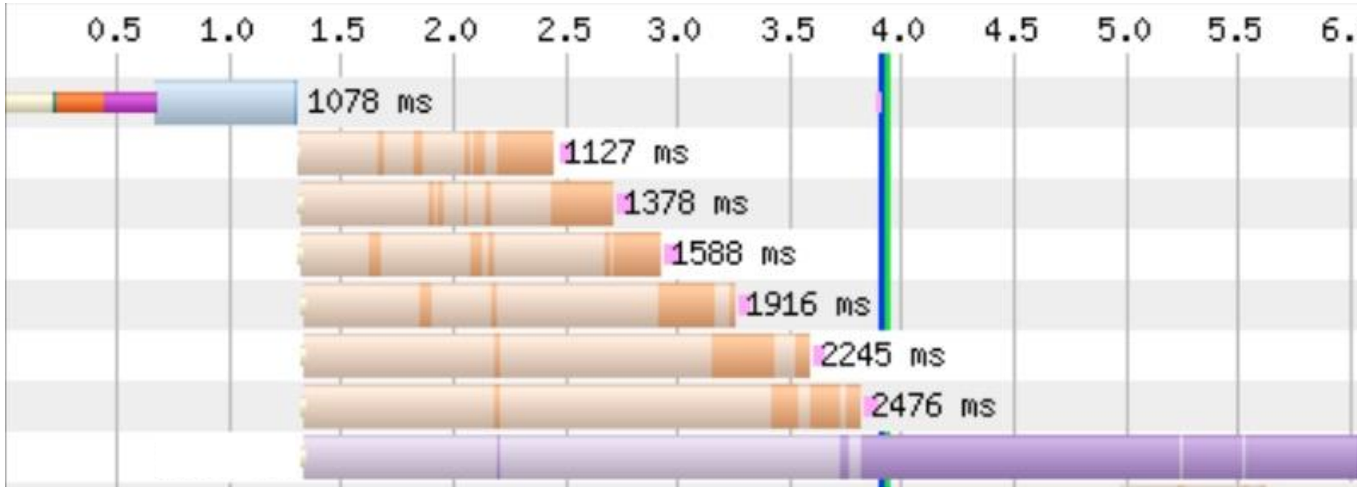
= loaded **after** parser-blocking JS

What will the waterfall look like for <img src=... fetchpriority=high />?



preload on **bottom** of <head> + fetchpriority = high

= loaded **after** parser-blocking JS



<img src=... fetchpriority=high />

= loaded **after** parser-blocking JS

You (probably) don’t need a preload if the image is in the HTML

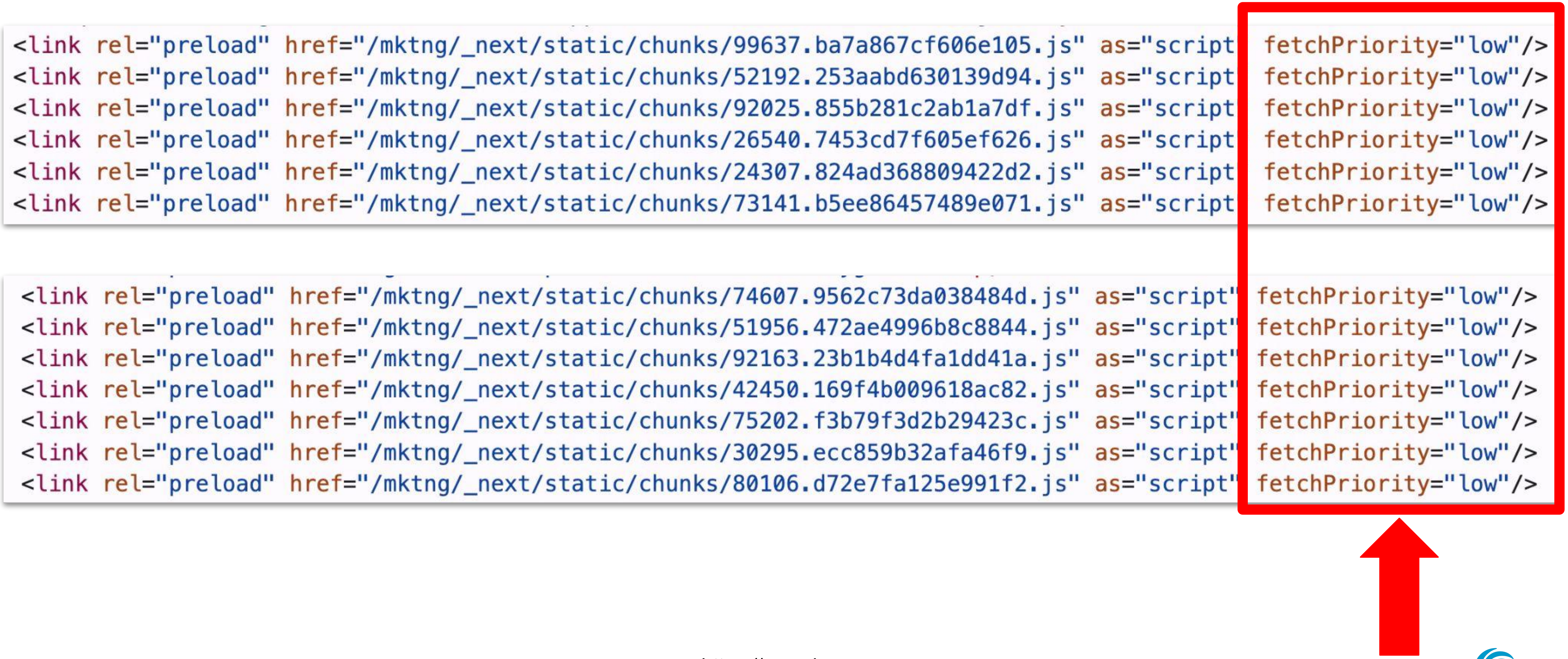
Preload should be applied with *surgical precision*

- Specific edge cases (you ***really*** know what you're doing)

- If the resource **isn't in the HTML**
  - Fonts
  - Dynamic LCP images
  - JS imports

https://web.dev/articles/preload-critical-assets

NEXT.js preloading 13 JS files needed for future navigations



```
<link rel="preload" href="/mktng/_next/static/chunks/99637.ba7a867cf606e105.js" as="script fetchPriority="low"/>
<link rel="preload" href="/mktng/_next/static/chunks/52192.253aabd630139d94.js" as="script fetchPriority="low"/>
<link rel="preload" href="/mktng/_next/static/chunks/92025.855b281c2ab1a7df.js" as="script fetchPriority="low"/>
<link rel="preload" href="/mktng/_next/static/chunks/26540.7453cd7f605ef626.js" as="script fetchPriority="low"/>
<link rel="preload" href="/mktng/_next/static/chunks/24307.824ad368809422d2.js" as="script fetchPriority="low"/>
<link rel="preload" href="/mktng/_next/static/chunks/73141.b5ee86457489e071.js" as="script fetchPriority="low"/>
```

```
<link rel="preload" href="/mktng/_next/static/chunks/74607.9562c73da038484d.js" as="script" fetchPriority="low"/>
<link rel="preload" href="/mktng/_next/static/chunks/51956.472ae4996b8c8844.js" as="script" fetchPriority="low"/>
<link rel="preload" href="/mktng/_next/static/chunks/92163.23b1b4d4fa1dd41a.js" as="script" fetchPriority="low"/>
<link rel="preload" href="/mktng/_next/static/chunks/42450.169f4b009618ac82.js" as="script" fetchPriority="low"/>
<link rel="preload" href="/mktng/_next/static/chunks/75202.f3b79f3d2b29423c.js" as="script" fetchPriority="low"/>
<link rel="preload" href="/mktng/_next/static/chunks/30295.ecc859b32afa46f9.js" as="script" fetchPriority="low"/>
<link rel="preload" href="/mktng/_next/static/chunks/80106.d72e7fa125e991f2.js" as="script" fetchPriority="low"/>
```

https://vercel.com

Other topics I researched

- 103 Early Hints
  - Tight mode impact?
  - Preloading responsive images?
- Why do font preloads need a crossorigin attribute?!?
  - Except on Safari?!!?!!!
  - Credentialed requests and CORS
  - Connection coalescing
- Tight mode across connections: chrome vs safari
- LCP load delay vs render delay
- Tight mode impact for Speculation Rules API (prefetch/render)
- *How much I hate browser devtools sometimes :)*

Ask me about these sometime ;)

M
L

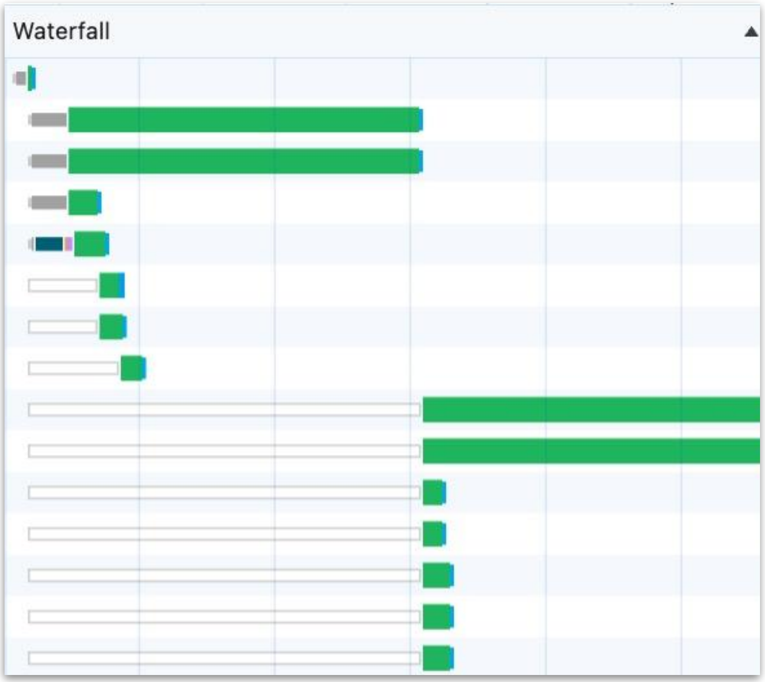
Network Performance isn’t the *most impactful* thing

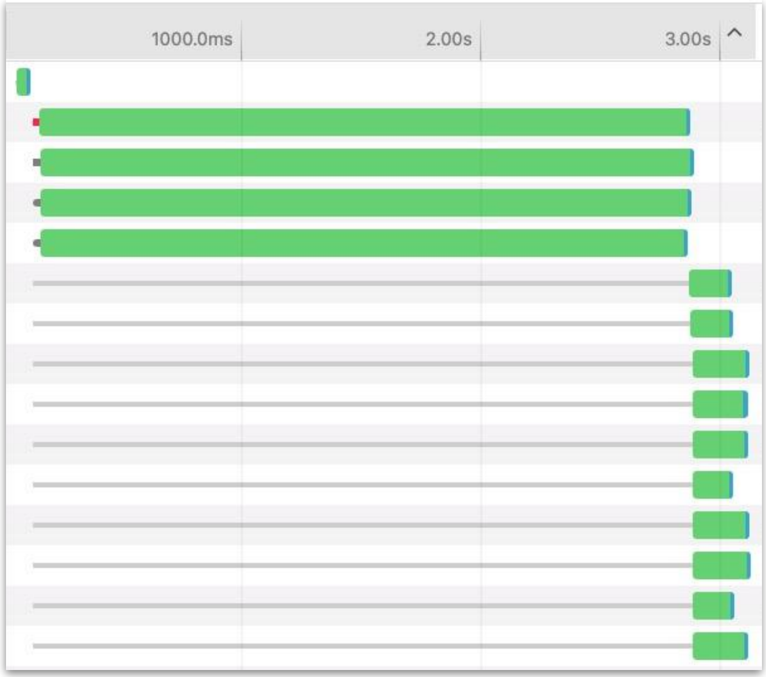
> “If you’re loading 5MB of JavaScript without a CDN, you have **bigger problems than just tight mode messing up!**”

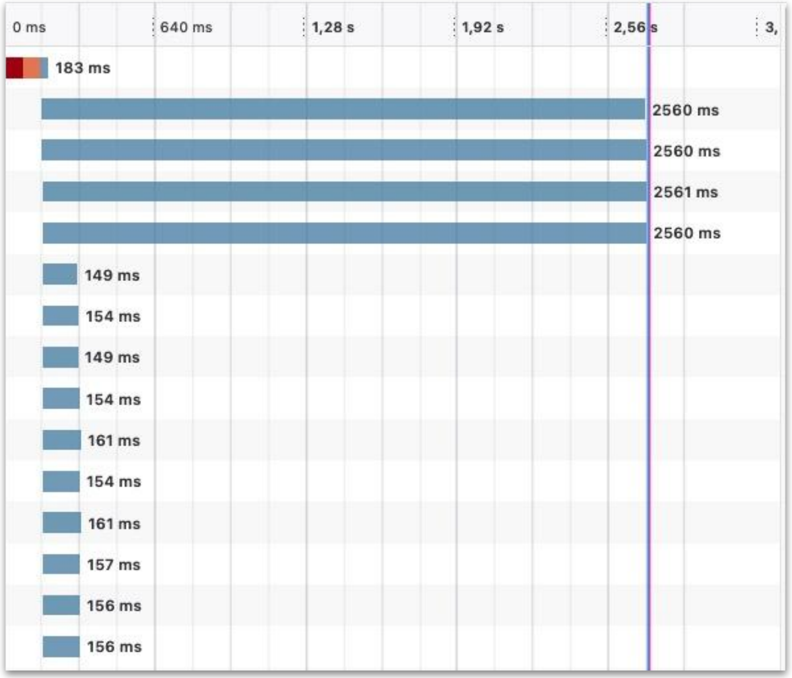
Robin Marx, *WeLoveSpeed 2024*

Exact same HTML,

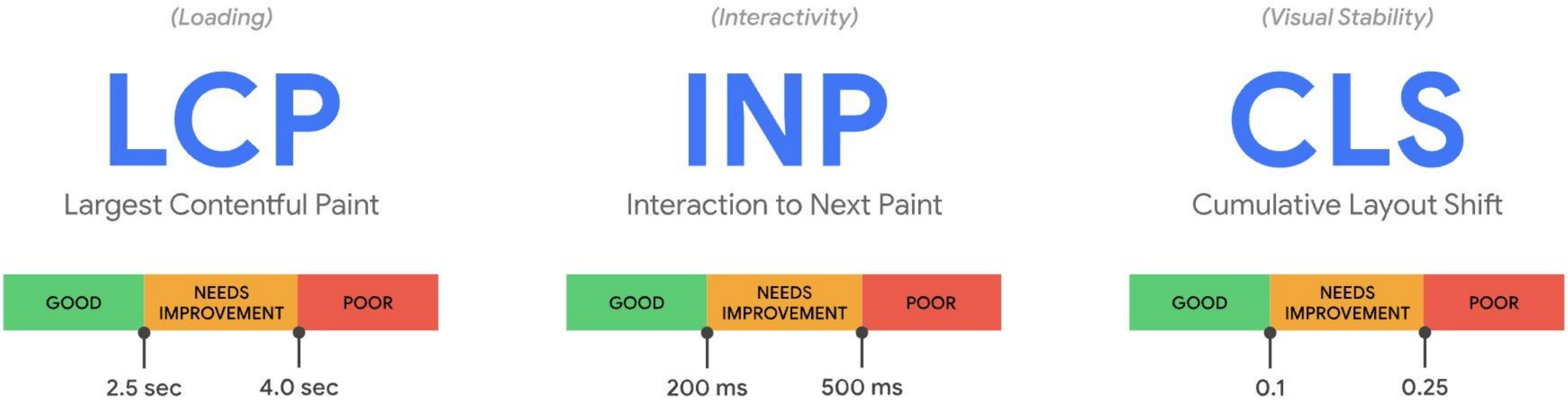
radically different behaviour







“Chrome” Web Vitals



(Loading)

LCP

Largest Contentful Paint

GOOD | NEEDS IMPROVEMENT | POOR

2.5 sec

4.0 sec

(Interactivity)

INP

Interaction to Next Paint

GOOD | NEEDS IMPROVEMENT | POOR

200 ms

500 ms

(Visual Stability)

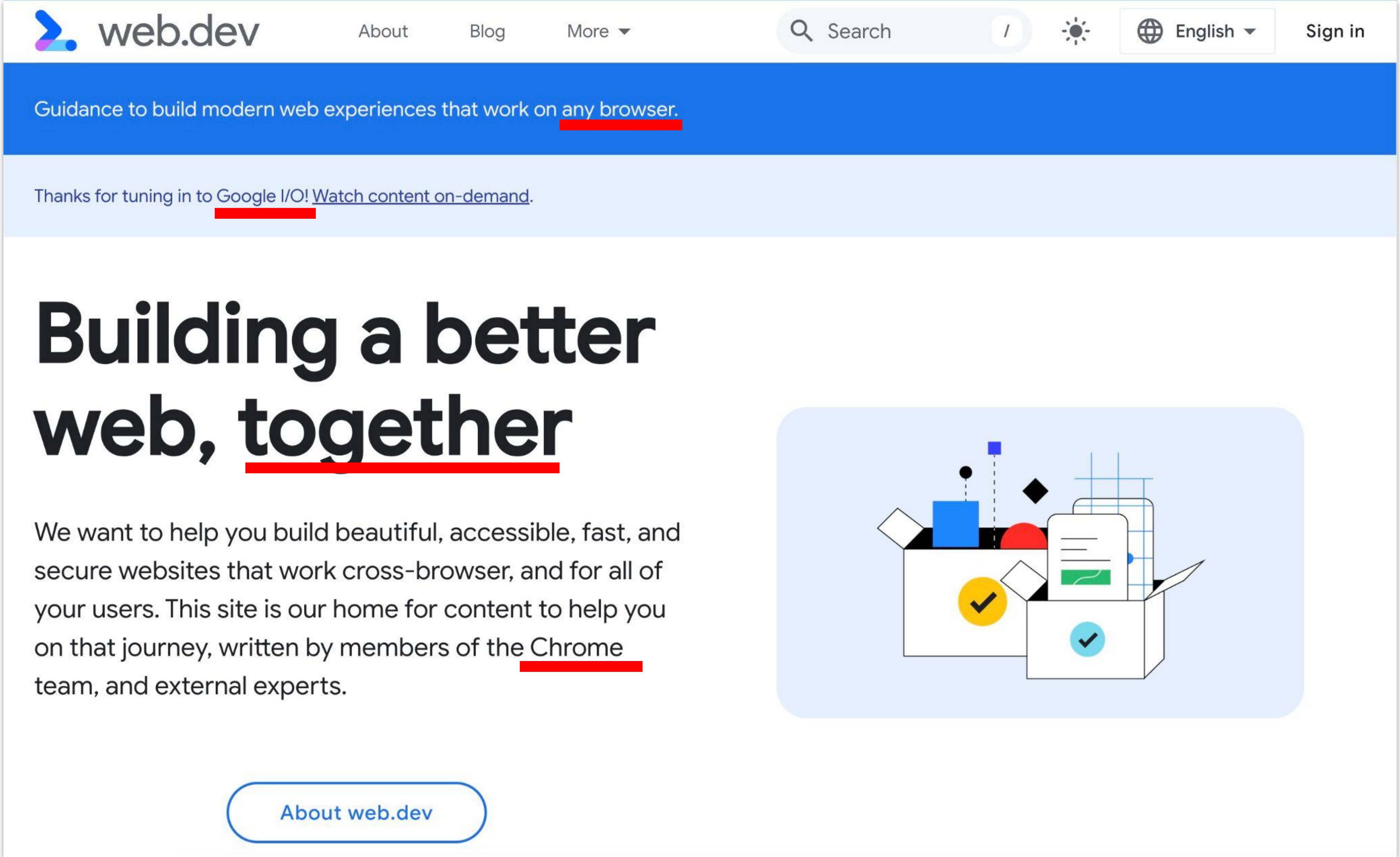
CLS

Cumulative Layout Shift

GOOD | NEEDS IMPROVEMENT | POOR

0.1

0.25

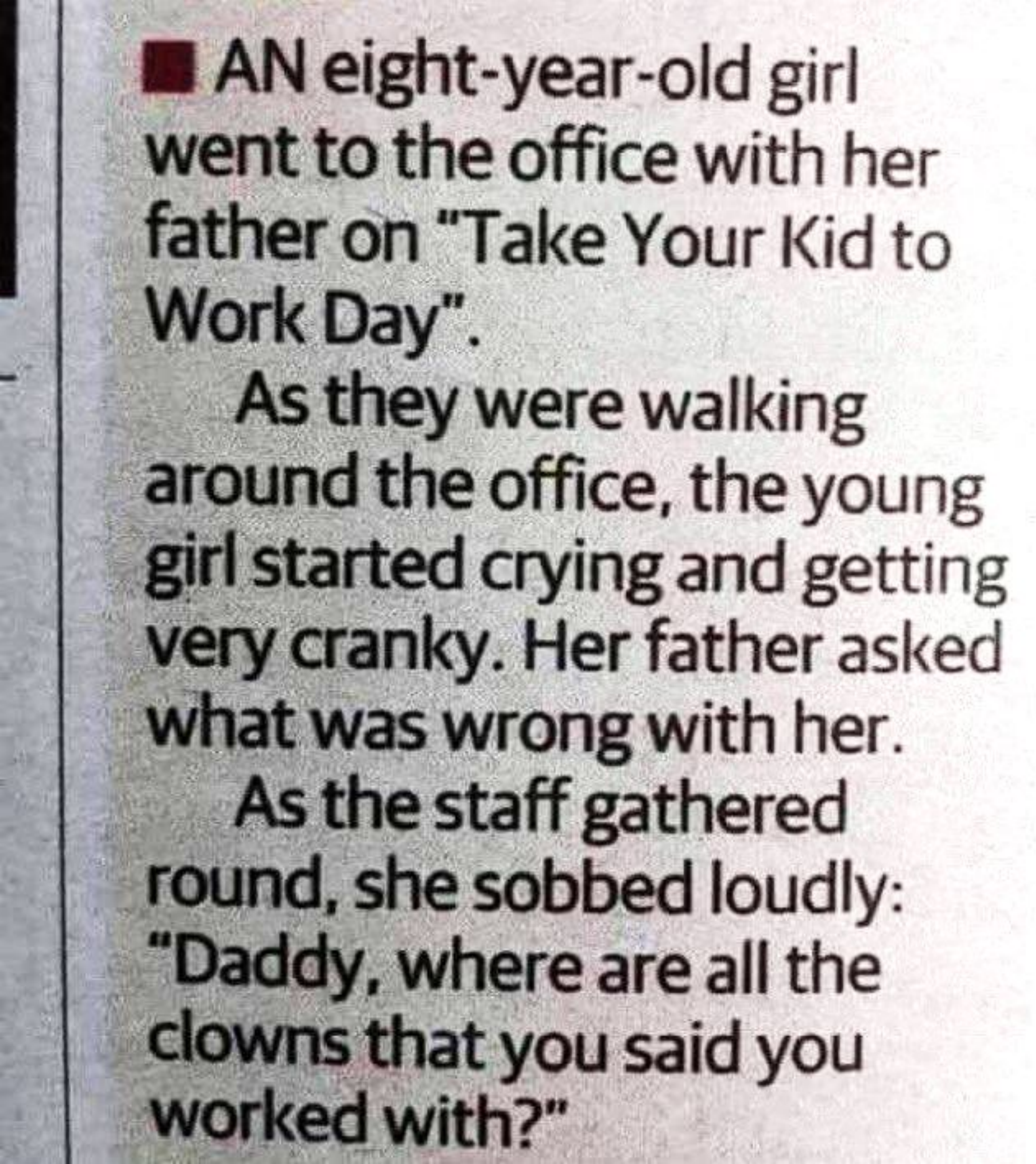


Guidance to build modern web experiences that work on any browser.

Thanks for tuning in to Google I/O! Watch content on-demand.

Building a better web, together

We want to help you build beautiful, accessible, fast, and secure websites that work cross-browser, and for all of your users. This site is our home for content to help you on that journey, written by members of the Chrome team, and external experts.



■ AN eight-year-old girl went to the office with her father on "Take Your Kid to Work Day".

As they were walking around the office, the young girl started crying and getting very cranky. Her father asked what was wrong with her.

As the staff gathered round, she sobbed loudly: "Daddy, where are all the clowns that you said you worked with?"

THANK YOU

THANK YOU