

Web [2] 3 : Full : hello-backbonejs - Google Chrome

hello-backbonejs

file:///home/haiswu/github-js/JSTraining/hello-backbone/index.html

Add list item

- Hello Backbone - Script 1
- Add list item
- Hello Backbone - Script 2
- Add list item
- Hello Backbone - Script 3
- Add list item
- Hello Backbone - Script 4

Lazy loaded very obviously thus gives a general hints that some resources could be loaded in this way in order to improve page performance

1). HTML parsing; should include the time downloading/execute external scripts therefore CDN and browser cache are helpful at this stage.

2). Execute DOMContentLoaded event scripts (includes jquery document ready handlers)

3). Fetch all dynamic scripts which created when DOMContentLoaded (plus other style, image resources but not at this case.)

4). Execute window onload event scripts

jquery-1.8.0 tar.gz

Elements Resources Network Scripts Timeline Profiles Audits Console PageSpeed

Search Network

Name Path	Method	Status Text	Type	Initiator	Size Content	Time Latency	Timeline
index.html	GET	Success	text/html	Other	0B	3ms	
jquery-1.8.2.js	GET	200 OK	applicati...	index.html:8	77.70KB	3.08s	
json2.js	GET	200 OK	applicati...	index.html:8	5.63KB	3.34s	
timing.js	GET	Success	applicati...	index.html:8	0B	16ms	
5.js	GET	Success	applicati...	index.html:8	0B	16ms	
underscore-min.js	GET	200 OK	applicati...	index.html:8	3.71KB	3.56s	
backbone-min.js	GET	200 OK	applicati...	index.html:8	4.21KB	3.78s	
1.js	GET	200 OK	text/plain	jquery-1.8.2.js:830	1.16KB	1.12s	
3.js	GET	200 OK	text/plain	jquery-1.8.2.js:830	1.25KB	1.08s	
4.js	GET	200 OK	text/plain	jquery-1.8.2.js:830	1.41KB	343ms	

10 requests | 95.06KB transferred | 6.99s (onload: 6.66s, DOMContentLoaded: 4.98s)

6.99s

6.66s

4.98s

(4+7+3863+6+1105)ms ≈ 4.98s

(1099+7+565)ms ≈ (6.66-4.98)s

timing.js:31  
timing.js:34  
timing.js:35  
timing.js:36  
timing.js:37  
timing.js:38  
timing.js:39  
timing.js:40  
timing.js:41  
timing.js:42

Show all downloads