H2 Server Push Performance

Measured over 11 days

June 14th - June 25th, 2018
Adaptive Acceleration

- Using RUM data to find critical resources
- Push those critical resources during the HTML generation “think time”
- Utilize idle network time
- Start “slow-start” earlier
HTTP/2 push is tougher than I thought

"HTTP/2 push will solve that" is something I've heard a lot when it comes to page load performance problems, but I didn't know much about it, so I decided to dig in.

HTTP/2 push is more complicated and low-level than I initially thought, but what really caught me off-guard is how inconsistent it is between browsers – I'd assumed it was a done deal & totally ready for production.

This isn’t an "HTTP/2 push is a douchebag" hatchet job – I think HTTP/2 push is really powerful and will improve over time, but I no longer think it’s a silver bullet from a golden gun.
How to interpret results

- Difference in DOMComplete time - negative/smaller is better
- Chrome-only, first-view results
- Difference is represented as a bar with a min and max values
- 95% confidence difference falls within the min and max values
- Green == Faster
- Red == Slower
- Blue == difference is not statistically meaningful
Grow revenue opportunities with fast, personalized web experiences and manage complexity from peak demand, mobile devices and data collection.
Grow revenue opportunities with fast, personalized web experiences and manage complexity from peak demand, mobile devices and data collection.

Confidence Interval : Mean

(-87.25,280.39) : 96.57
(-284.14,217.57) : -33.28
(-2800.31,-110.41) : -1455.36
(-795.92,-300.42) : -548.17
(-271.58,-141.74) : -206.66
(-1092.97,-659.11) : -876.04
(-160.8,168.75) : 3.97
(-181.47,168.79) : -6.34
(-851.39,226.46) : -312.46
(-534.9,30.18) : -252.36
(-423.6,-185.64) : -304.62
(-663.12,291.28) : -185.92
(-288.51,-77.94) : -183.23

Desktop A2 Pushed Performance Overview

improvement

- no statistical difference
- yes

Pushed On Effect (ms)

8
Observations

● Of the results where there is a statistical difference, we are seeing positive improvement with H2 Server Pushes applied.

● Mobile performance data has more variation and noise, resulting in higher uncertainty (longer bars).
Statistical Methodology

● Linear regression methodology used for statistical calculations, based on the following dimensions:
  ○ Geographic location
  ○ Client OS
  ○ Browser user-agent
  ○ Hour of day
  ○ Day of week
  ○ ISP
  ○ URL

● This allows us to statistically account for natural performance variations due to the impact of the above dimensions
Statistical Methodology

- The set of customers used are those that:
  - Have enabled automated Server Push on Akamai
  - Are also using the new mPulse RUM product
  - Generate enough traffic for the statistical computation

- The traffic that was considered for this computation was restricted to:
  - Chrome only (split into Desktop and Mobile)
  - Restricted to “first-view” requests
  - Top 10 URLs for the customer web site in terms of traffic volume
  - Minimum of 1000 hits for a URL to be eligible
  - At least 10% of the URL hits must have had H2 Server Pushes applied
What’s next?

- Will continue to gather data as the set of customers expands
- Will report back