Chrome Interventions Quarterly

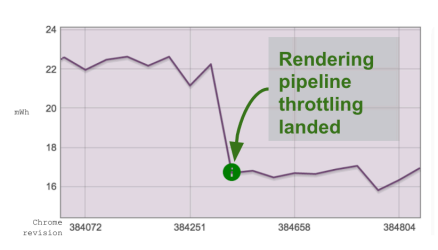
"One of the greatest quarterly of our time" — *Kenji West* ;p

2016 Q2~Q3

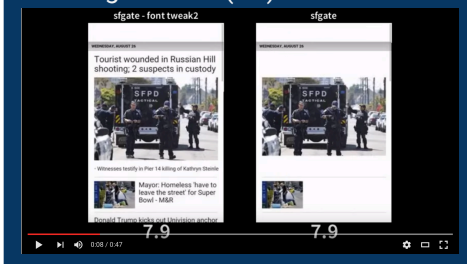
WICG github for cross vendor discussion
Public ML: intervention-dev@chromium
Public backlog of interventions

We need you! Contact ojan@chromium and kenjibaheux@chromium to get involved.

Rendering pipeline throttling
~27% win on power consumption for tough ad cases:



Web font intervention
2.5 seconds win on Time To First Meaningful Paint (3G):



Shipped and trialed interventions

Rendering pipeline throttling for offscreen cross-origin frames 52

Motivation: sophisticated third party content can be quite taxing on CPU and battery.
Solution: avoid unnecessary work by pausing the rendering pipeline for offscreen cross-origin frames. **"Frees up CPU cycles, reduces power consumption... Awesomeness defined!"**.
Status: shipped in Chrome 52.

Disallow pop-ups from cross-origin frames on non-tap touch events 52

Motivation: abusive third party content that open a popup on a scroll are annoying.
Solution: refine the required "user gesture" for window.open/etc. to avoid these unintended pop-ups. **"Better user experience is like… better is better."**
Status: shipped in Chrome 52.

Unintervention: re-instate autoplay for muted/silent videos 53

Motivation: a while back autoplay was disabled on Android. Unfortunately, this decision didn't account for how the ecosystem would react (e.g. heavy JS video decoders)
Solution: relax the intervention, first by allowing muted/silent videos to autoplay.
Status: shipped in Chrome 53.

Bail out on web fonts with an effectively slow connection 53 `trial`

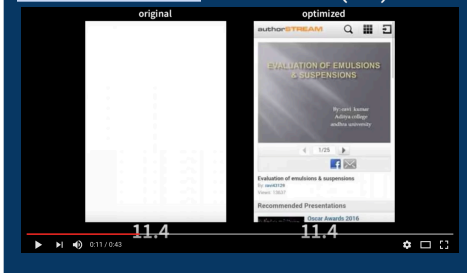
Motivation: Chrome waits up to 3 seconds for any given web font. Turns out, on slow connections, most web font requests actually hit the timeout :(
Solution: extend the 2G intervention we shipped in 49 to effectively 2G-like connections as determined by the network quality estimator. **"Sometimes one has to stop being emotional over fonts."**
Status: field trial in Chrome 53; waiting for data.

Shipping real soon

Throttle the JS timers of offscreen frames 54~

Motivation: sophisticated third party content can be quite taxing on CPU and battery.
Solution: avoid unnecessary work by throttling JS timers for offscreen frames.
"Awesomeness! We already established that."
Status: aiming at 54.

Document.write intervention
18 seconds win on TTFMP (2G):

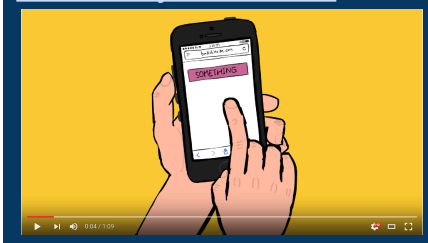


Blocking perf-taxing document.write for users on 2G 54~

Motivation: for users on slow connections, scripts loaded via document.write can delay display of main page content for tens of seconds, or even lead to page load timeouts.
Solution: blocking perf-taxing document.write statements delivers dramatic speedups and helps improve the web by incentivizing the use of faster asynchronous alternatives.
"Like a speedboat, I tell y'all!"

Status: aiming at 54 after a highly successful field trial in 52: 25% more successful page loads, 38% faster mean time from parse start to parse completion (5+ seconds win)...

What annoys on the web



lmma go for what annoys, how y'all like that?

Upcoming interventions to address what annoys on the web (see sidebar)

Scroll anchoring

Motivation: above the fold elements loading push down what's visible on screen.
Solution: anchor the scroll position to what's visible at the top of the screen.
Status: refining approach; aiming at a field trial in M55~
Dogfood: **chrome://flags/#enable-scroll-anchoring;** send feedback to g.co/reportbadreflow

Disable the Vibrate API for cross origin frames

Motivation: the Vibrate API is abused to trick users to fall for social engineering schemes (about half of the websites using Vibrate API appears to be categorized as Social Engineering by Safe Browsing)
Solution: start with disabling the Vibrate API for cross origin frames.
Status: implementing.

Only allow navigations/popups from cross-origin frames on click

Motivation: abusive third party content that navigate the top level page on a scroll are an annoyance.
Status: data came back, will proceed with a field trial.

History back that works!

Motivation: sites that navigate you forward after you hit back :/
Solution: adding to the history requires a user gesture if arrived via history navigation.
Status: proposal

Ignore clicks on iframes that have moved in the last XXms

Motivation: unintentional clicks on elements that just moved is not uncommon.
Status: design doc; aiming at building a prototype in Q3.

Project OldSpice: taming annoyances with JS dialogs

Motivation: abusive JavaScript dialogs
Solution: cancel modal dialogs on tab switch & disallowed them on background pages
Status: implementation started.

Yo signing off... I gotta go battle perf issues

Other upcoming interventions motivated by performance issues

Turn Document Level Event Listeners into passive ones

Motivation: touch/wheel event handlers can cause major hiccups to user scroll intent.

Solution: explore if/how we could stop blocking on these events for the case where the event listener is on the root element.

Status: experiment in dev/canary (M54)

Make touchstart during a fling uncancelable

Motivation: blocking touchstart handlers during fling can cause major hiccups during fling boosting / flywheel fling.

Solution: explore if/how we could stop blocking on these events.

Status: implementing; aiming at a field trial M55~

Lower loading priority / or not loading offscreen cross origin iframes

Motivation: minimize the workload from offscreen elements

Solution: explore if/how we could lower the loading priority or not even load cross origin iframes until they are about to come into view.

Status: landing metrics to better understand the pros and cons.