

# Performance Comparison of Browsers

# Motivation

- Updating Firefox to improve its performance (over time)
- How to know if users actually see improvements?
- Additionally, comparison with other browsers such as Chrome (or based on Chromium), Safari, ...
- Therefore, need for some tool that allows capturing content of webpages and replay the page loads under emulated conditions

# Vision

- Collect information on paths and timings from first DNS request up until the last transferred byte
- Transform these records to repeatable test cases
- Allows to simply run in CI and compare, e.g., different browser code/versions or protocol tweaks individually
  - Determine whether new proposals help the average user

# Challenges

- How to accurately record things for comparison to be representative?
- Webpages change over time
- Content interdependencies
- Different network conditions (paths, congestion, losses, 3G, 4G, ...)
- Browsers work differently (browser pipelines/logic, rendering, ...)
- Changes to protocols and stacks
- What to optimize for? 3G? 4G?
  - Optimize for common cases vs boundaries?

# Naive Solutions

- Replay whole traffic to mimic pageload
  - Problem: Security features of protocols make this difficult
  - Idea: Simply remove all security features?
  - Still need to rewrite connection IDs in QUIC; deviations in packets will cause issues
- Fetch all objects from a website (e.g., curl, wget) and store them locally, then replay everything locally under different conditions
  - Problem: Objects come from different servers
  - Idea: Treat different URLs/IP addresses as different paths?

# Existing Solutions

- Mahimahi, C++  
<https://www.usenix.org/conference/atc15/technical-session/presentation/netravali>
  - However, does not perform well with high throughputs (Gigabit)
- Catapult (Google)  
<https://chromium.googlesource.com/catapult/>
  - fetches objects, then allows replay
- Google Lighthouse, Project Lantern  
<https://github.com/GoogleChrome/lighthouse/blob/main/docs/lantern.md>
  - Creates dependency tree, simulates behavior under different conditions
  - Mostly targeted at Web devs to check if their pages load quickly

# Open Question

- Where to go from here?
- One software for everyone that includes multiple different browsers?  
Or share learning but keep building/maintaining individual setups?
- A lot of Mozilla (longitudinal) data publicly available  
<https://dictionary.telemetry.mozilla.org/>  
<https://glam.telemetry.mozilla.org/>
- Also: simply ask for data dumps!