Modeling Web Page Performance Improvements via Loading Graphs and Testbed Simulations

We want to investigate whether we can improve real-world web page loading times by switching the API object serializer from JSON to CBOR [1]. However, which method allows us to find this out? This thesis shall investigate a novel approach to model and simulate this improvement. On one side, we can crawl popular websites and measure the loading times. This creates a lot of data that we have to simplify, e.g., by creating an acyclic loading graph like [2]. On the other hand, we can use performance measurements in our testbed to measure improvements using different technologies. However, this improvement might be irrelevant in the total time a web page loads. To get a real-world perspective, we can combine our model of the loading time and our performance measurements to estimate the overall improvement.

- You develop a specialized crawler (e.g., based on Selenium or Puppeteer [3, 4]), that loads a web page and generates a loading graph similar to [2]
- The crawler should store any JSON responses together with the loading time metrics
- With this data, you model the network behavior of the JSON requests (also considering cached and uncached responses)
- On our testbed, you use the crawled JSON responses and the observed bandwidth to simulate the request using both JSON and CBOR.
- Now, the simulated performance can be applied to the loading graph to estimate how long the page would have loaded using the different technology

Good understanding how data is transferred on the Internet, knowledge of the Linux network stack, Bash, Javascript

Markus Sosnowski  sosnowski@net.in.tum.de
Florian Wiedner  wiedner@net.in.tum.de

References
[1] https://cbor.io/