Cloud is not a silver bullet: A Case Study of Cloud-based Mobile Browsing

Ashiwan Sivakumar, Vijay Gopalakrishnan, Seungjoon Lee, Sanjay Rao, Subhabrata Sen and Oliver Spatscheck

Purdue University, AT&T Labs - Research

Presenter: Saurabh Verma
• In real world does cloud browsers (CB) actually provide benefits that they are meant for?

• Does CB actually save energy and provides fast browsing? And to what extent!!

• Need a systematic understanding in the design space that involves cloud and browser.
Investigates CB performance against normal browsers (direct).

Provides scenarios when CB fails.

Provides some understanding of the rich design of cloud browser.
How CB differs from direct?

Cloud Browsers (cont.)

CB provides many additional functionality to users:

- JS execution.
- Energy Saving.
- Data compression.
- Transcoding images.
- Traditional benefits of having cloud-like storage, sharing etc.
Cloud Browsers (cont.)

Request web page by creating a persistent connection.

Cloud fetches data from http site on behalf of user.

Cloud executes JS, compress data, resize images and send CBML to user.

Cloud continuously executes JS and pushes burst of objects whenever page changes.
Network Activity

**Direct**
- Parallel HTTP connection.
- JS Processing
- fetch objects from web server

**CB**
- Wait Time includes Parsing, JS Processing, fetch objects from web server.
- Serves CBML format pages to client.
Evaluation Goal and Metrics

**Goal**
- Quantify the impact of offloading JS execution to the cloud.
- Quantify the benefits of data compaction of pages in cloud.

**Definitions of Metrics**
- **Page download time**: Time between the first SYN and last ACK for all objects in a page observed on the device.
- **Total Energy**: Sum of 1) CPU energy consumed by a browser 2) Network Energy.
Experiment Setup

Disable cache on both browsers

1. Calculate
   1) Download time difference.
   2) Energy difference.

2. Calculate
   1) Download time difference.
   2) Energy difference.

Cloud Browser

Motivation

Intro.

Goals

Network
Results: Impact of offloading JavaScript execution

Average delta (CB time – direct time) for each page with CB.

P(CB time – direct time <0) = 0.62
P(CB time < direct time) = 0.62
P(CB time > direct time) = 0.38

CB decreases the download time compared to Direct for 38% of pages. ?? Did authors made a mistake. Help!!

But CB can also increase download time up to 29.8sec for other pages.
Similar Results With Energy as well....

(CB Energy – direct Energy)

Scatter Plot
What Results say?

- Energy consumed by 52.7% pages
- Energy consumed by 48.3% pages

Result JS
Result Energy
CB fails
Result session
Data comp.
Conclusion
Why CB fails to decrease download time or save energy for some pages?

Extent of JS in the page

- Heavy JS pages are okay. But for some pages with light JS processing-- does not help in saving energy. Decompressing CBML outweighs the saving obtained from running JS in cloud.

Long JS vs short JS

- Long-running JS pages are bad because CB may result in having data compaction ratio > 1. Cloud sends multiple burst of data containing-- which diminishes the data compression efficiency due to overheads involves in sending each burst of data.
Why CB fails to decrease download time or save energy for some pages?

Any suggestions
How CB can save energy for long running JS pages?

Hint: common techniques used for storing/compressing buffering data.
Enable cache everywhere!!

Scroll images inside a webpage. Note down energy increase and download time
Energy usage over user session

Direct cache JS along with images. So no further network energy required.
CB needs to run JS on cloud each time user clicks. Then cloud sends CBML containing particular image.
Impact of Data Compaction

CB increases the total energy for 83% of pages which do not have JS script embedded (No JS Total) as compare to Direct.

As compared to No JS Total CB increases energy for 47.3% pages for the JS pages (JS Total).

CDF of energy with CB

Result JS
Result Energy
CB fails
Result session
Data comp.
Conclusion
Impact of Data Compaction (cont.)

CDF of data compaction ratios are smaller when JS is excluded from the pages, explaining the higher increase in energy with CB when compared to the version with JS.

Data compaction ratio is > 1. CB might send page layout information in the CBML to aid the page rendering in the client and this might be an additional overhead for small pages.
Impact of Data Compaction (cont.)

Another key reason for more energy consumption: CB proxy sometime takes a long Wait Time to download the page from the server and perform data compaction (even if its 0.09), thereby increasing the total download time, energy compared to Direct.

Possible design decision: Avoid data compaction when wait time is large. Saving time in data compaction can help in reducing network energy.

CDF of data compaction ratios

CDF

Data compaction ratio (CBSize/DirSize)

0 0.2 0.4 0.6 0.8 1

0 0.5 1 1.5 2 2.5

With JS
Without JS

Result JS Result Energy CB fails Result session Data comp. Conclusion
Lessons Learned

- Offloading JS to the cloud is not necessarily beneficial.
- Considering user interactivity when offloading JS is important.
- Data compaction is not always beneficial.