Why are Web Browsers Slow on Smartphones?

Zhen Wang (Rice)
Felix Xiaozhu Lin (Rice)
Lin Zhong (Rice)
Mansoor Chishtie (TI)
WINDOW TO THE CLOUD
Mobile browsers are slow

- Nexus One (N1) 12 sec
- HTC Dream (G1) 17 sec

Top 10 Mobile Website
Top10 Non-mobile Webpage
• What does the browser show?

• How does the browser work?

• Where is the bottleneck?
What does the browser show?

Non-mobile webpages  Mobile webpages

avg
How does the browser work?

**IR operations:** Parsing, Style, Scripting, Layout, Painting
Where is the bottleneck?

• Existing work on PC browsers
  – Layout
  – Style formatting
  – Scripting
Is it true for mobile browsers?

Layout, Style, Scripting
Performance characterization

• Metric: *browser delay*
  – *Starting point*: when the user presses the “GO” button of the browser to open an URL.
  – *End point*: when the browser’s page loading progress bar indicates 100%.
Performance characterization

• Dependency timeline characterization

• What-if analysis
Dependency timeline characterization

- **Resource Loading**
- **Parsing**
- **Scripting**
- **Style**
- **Layout**
- **Painting**

Resource Group

Elapsed time (ms)

Request
Dependency timeline characterization

![Graph showing resource loading, parsing, scripting, style, layout, and painting over elapsed time (ms).]
Dependency timeline characterization

Elapsed time (ms)

Resource Group

Data packets

Resource Loading
Parsing
Scripting
Style
Layout
Painting

Resource Loading
Parsing
Scripting
Style
Layout
Painting

Elapsed time (ms)
Dependency timeline characterization

This is part of resource loading. It is mainly spent on network.
Glue operation

Dependency timeline characterization

- Resource Loading
- Parsing
- Scripting
- Style
- Layout
- Painting

Resource Group

Elapsed time (ms)
Dependency timeline characterization

- Resource Loading
- Parsing
- Scripting
- Style
- Layout
- Painting

*intra-group dependency*
Dependency timeline characterization

Inter-group dependency
What overall performance gain will be achieved if a browser operation is accelerated?
What-if analysis

Elapsed time (ms)

Resource Group

- Resource Loading
- Parsing
- Scripting
- Style
- Layout
- Painting

Shrink
What-if analysis

Elapsed time (ms)

Resource Group

0              1000            1500            2000            2500            3000            3500            4000

1  2  3  4

Shift to the left

0  1  2  3  4  5  6  7  8

Resource Group

Elapsed time (ms)

Resource Loading  Parsing  Scripting  Style  Layout  Painting

Shift to the left

Shrink
What-if analysis

- Resource Loading
- Parsing
- Scripting
- Style
- Layout
- Painting

Shift to the left
Shrink
They will not move

Elapsed time (ms)

Resource Group

Resource Group

0              1000            1500            2000            2500            3000            3500            4000
What-if analysis

Elapsed time (ms)

Resource Group

0              1000            1500            2000            2500            3000            3500            4000

Resource Loading Parsing Scripting
Style Layout Painting
What-if analysis

![Graph showing resource loading, parsing, scripting, style, layout, and painting times over elapsed time (ms).]

- Resource Loading
- Parsing
- Scripting
- Style
- Layout
- Painting

Original
What-if analysis

Resource Group

Elapsed time (ms)

Resource Loading  Parsing  Scripting  Style  Layout  Painting

New  Original
Experimental setup

• Platform:
  – HTC Dream (G1): 528MHz
  – Nexus One (N1): 1GHz

• Operating System
  – Android 2.1 (Eclair)

• Benchmark Webpages:
  – Top 10 mobile websites
  – Top 10 visited non-mobile webpages from LiveLab
Experimental setup

• We used three network conditions:
  – Emulated enterprise Ethernet (no traffic control)
  – Typical 3G network (T-mobile)
  – Emulated adverse network
    • First-hop RTT: 400ms
    • Bandwidth (downlink/uplink): 500Kbps/100Kbps
Logging information

• Time stamp for browser operations
  – Overhead: <1%

• Tcpdump
  – Overhead: <2% (CPU); <0.4% (MEM)
Results

two take-away messages
IR operations do not matter much!

Parsing, Style, Scripting, Layout, Painting
IR operations do not matter much

- **Layout Calculating Speedup**
  - Non-mobile Web
  - Mobile Web

- **Style Formatting Speedup**
  - Non-mobile Web
  - Mobile Web

- **Scripting Speedup**
  - Non-mobile Web
  - Mobile Web
IR operations do not matter much

### Layout Calculating Speedup
- Non-mobile Web
- Mobile Web

### Style Formatting Speedup
- Non-mobile Web
- Mobile Web

### Scripting Speedup
- Non-mobile Web
- Mobile Web
IR operations do not matter much

Combined: Parsing, Layout, Style, Scripting, Painting, Glue

Overall Improvement

Speedup

- Non-mobile Web
- Mobile Web
Resource loading is the bottleneck!
Resource loading is the bottleneck

![Graph showing resource loading improvement for non-mobile and mobile web]

Overall Improvement

Resource Loading Speedup

- Non-mobile Web
- Mobile Web

600%
Resource loading is the bottleneck

![Graph showing resource loading speedup for non-mobile and mobile web.](image)

- Overall Improvement vs Resource Loading Speedup
- Non-mobile Web
- Mobile Web
- 600% improvement
Resource loading is the bottleneck

- Network RTT
- Network Bandwidth
- Browser loading procedure
- Processing power
Network RTT matters

- G1 - Non-mobile Web
- N1 - Non-mobile Web
- G1 - Mobile Web
- N1 - Mobile Web

Browser Delay (sec) vs Injected RTT (ms)

- 3G
Network bandwidth doesn’t matter

![Graph showing browser delay vs. bandwidth for non-mobile and mobile Web under different conditions](image)

- **G1 - Non-mobile Web**
- **N1 - Non-mobile Web**
- **G1 - Mobile Web**
- **N1 - Mobile Web**
Browser loading procedure

New resources can only be discovered after parsing a loaded resource.
Browser loading procedure

Redirections on the main HTML file further delay the discovering time of later resources.
Browser loading procedure

Parsing of the HTML file is blocked by JavaScripts
Browser loading procedure

- Up to **25** concurrent requests for top mobile/non-mobile webpages

- Constrains on concurrent TCP connections

<table>
<thead>
<tr>
<th>Mobile Browser</th>
<th>Connections/hostname</th>
<th>Maximum connections</th>
</tr>
</thead>
<tbody>
<tr>
<td>Android</td>
<td>4</td>
<td>4</td>
</tr>
<tr>
<td>iPhone 4.3</td>
<td>6</td>
<td>35</td>
</tr>
<tr>
<td>Blackberry 9700</td>
<td>4</td>
<td>16</td>
</tr>
<tr>
<td>Opera Mobile</td>
<td>4</td>
<td>4</td>
</tr>
<tr>
<td>Opera Mini</td>
<td>10</td>
<td>60</td>
</tr>
</tbody>
</table>

http://www.browserscope.org/
Average number of resources

- Top 10 Mobile Website: 22 resources
- Top 10 Non-mobile Webpage: 96 resources
Average number of network round trips

- Top 10 Mobile Website: 19
- Top 10 Non-mobile Webpage: 27
Processing power in resource loading

This is part of resource loading. It is *mainly* spent on network.

http://mail.yahoo.com
Processing power in resource loading

Elapsed time (ms)

Resource Group

DNS lookup query

http://mail.yahoo.com
Processing power in resource loading

Graph showing the processing power in resource loading with various groups and the corresponding elapsed time in milliseconds. The timeline includes steps such as DNS lookup query, TCP connection established, HTTP GET sent out, and stages of resource loading: Parsing, Scripting, Style, Layout, and Painting.
Processing power in resource loading

- Resource Loading
- Parsing
- Scripting
- Style
- Layout
- Painting

(i) DNS lookup query
(ii) TCP connection established
(iii) HTTP GET sent out

Resource Group

Elapsed time (ms)

http://mail.yahoo.com
Time spent by G1 and N1 for those three cases

(i) 126ms

(ii) 40ms

(iii) 98ms

G1 N1
Total time spent in the three cases on average when opening a mobile webpage

- G1: 2 sec
- N1: 1 sec
Processing power in resource loading

• Other uncategorized processing

  – The OS moves the data from network stack to browser after receiving data packets

  – Computation for secure connection (HTTPS)
More powerful hardware improves the browser delay mainly through faster OS services and network stack instead of faster IR operations.
Performance characterization results

• IR operations do not matter much

• **Resource loading is the bottleneck**
  – Network RTT (X)
  – Network bandwidth
  – Browser loading procedure (X)
  – Processing power (X)
How to improve mobile browser’s performance?

Reduce RTT

• Cloudlet
• Data staging

Reduce # of Round Trips

• Web Pre-fetching
• Resource batching
• Data URI scheme
• Speculative resource loading

On-going work

- Speculative mobile browser design
- Fully understand the impact of hardware
- OS and network service acceleration

http://www.owlnet.rice.edu/~zw3/projects_Tempo.html
## Benchmark webpages

<table>
<thead>
<tr>
<th>mobile</th>
<th>Non-mobile</th>
</tr>
</thead>
<tbody>
<tr>
<td><a href="http://www.espn.com">http://www.espn.com</a></td>
<td><a href="https://owlspace-ccm.rice.edu/portal">https://owlspace-ccm.rice.edu/portal</a></td>
</tr>
<tr>
<td><a href="http://mail.yahoo.com">http://mail.yahoo.com</a></td>
<td><a href="http://arstechnica.com/">http://arstechnica.com/</a></td>
</tr>
<tr>
<td><a href="http://mail.google.com">http://mail.google.com</a></td>
<td><a href="http://houston.craigslist.org/">http://houston.craigslist.org/</a></td>
</tr>
<tr>
<td><a href="http://mail.aol.com">http://mail.aol.com</a></td>
<td><a href="http://registrar.rice.edu/">http://registrar.rice.edu/</a></td>
</tr>
</tbody>
</table>