Dissecting Web Latency in Ghana

Yasir Zaki, Jay Chen
New York University Abu Dhabi, UAE
yasir.zaki@nyu.edu
jchen@cs.nyu.edu

Thomas Pötsch
University of Bremen
Bremen, Germany
thp@comnets.uni-bremen.de

Talal Ahmad, Lakshminarayanan Subramanian
CS, NYU and CTED, NYUAD
{ahmad, lakshmi}@cs.nyu.edu

ABSTRACT

Web access is prohibitively slow in many developing regions despite substantial effort to increase bandwidth and network penetration. In this paper, we explore the fundamental bottlenecks that cause poor web performance from a client's perspective by carefully dissecting webpage load latency contributors in Ghana. Based on our measurements from 2012 to 2014, we find several interesting issues that arise due to the increasing complexity of web pages and number of server redirections required to completely render the assets of a page. We observe that, rather than bandwidth, the primary bottleneck of web performance in Ghana is the lack of good DNS servers and caching infrastructure. The main bottlenecks are: (a) Recursive DNS query resolutions; (b) HTTP redirections; (c) TLS/SSL handshakes. We experiment with a range of well-known end-to-end latency optimizations and find that simple DNS caching, redirection caching, and the use of SPDY can all yield substantial improvements to user-perceived latency.

Categories and Subject Descriptors

C.2.2 [**Computer-Communication Networks**]: Network Protocols; C.4 [**Performance of Systems**]: Measurement techniques

Keywords

Web; HAR; DNS; Developing countries

1. INTRODUCTION

Web access in developing regions suffers from a continually widening content-connectivity gap: the rate of growth in connectivity is outstripped by the growth in the complexity of web pages. Despite lagging broadband and latency statistics relative to the Organisation for Economic Co-operation and Development (OECD) countries, Internet infrastructure in developing regions has been improving steadily over the past decade [1]. For example, Accra (the capital of Ghana) has several major undersea cable landing sites including: ACE, Main One, Glo1, and WACS [5]. Although this appears promising, the web has seen a 30 to 50-fold increase in average page complexity over the same decade [4, 11, 15, 19], which significantly outpaces the growth in connectivity.

Recent works have explored facets of slow Internet in developing regions from different perspectives, geographic locations, and infrastructural contexts. Some works find that high latency, buffer bloat, and packet loss are the central causes of poor performance [25]. Other works examine problems caused by the interaction between low bandwidth links and high network contention by TCP flows [16]. Yet other research cites routing protocols and architectural issues such as Content Distribution Networks (CDN) server placement among the central problems [4].

In this paper, we aim to understand the causes of user-perceived web latency in Ghana as it represents one of the most well connected developing countries in the world [5]. Unlike prior work in developing regions [16, 18, 25] that focus on access link performance in developing regions, our measurements detail the core infrastructural issues that lead to a poor web browsing experience. A number of works study similar web performance issues in fixed/wired or mobile environments such as [14, 31], but our unique context results in a different set of issues.

In our measurement driven approach, we collected download traces of Alexa's [2] top webpages from different locations in Ghana from 2012 to 2014. We also collected similar measurements in three different well-connected locations around the world over the same time period for comparison. We found the following key results: bandwidth is less of a bottleneck than several other factors, the actual time spent downloading content represents only a small fraction of the end-to-end page download time, and the main culprit of user-perceived web latency is actually DNS resolution. Specifically, DNS caching along with placement of the root and top level domain DNS servers play the most significant role in the overall web performance. We find that caching DNS records can decrease page load times in Ghana by five times. We also observe that other important factors for high end-to-end latency include HTTP redirections and TLS/SSL handshakes for secure pages. Caching HTTP redirects can enhance the page load time by 20%, and long TLS/SSL handshakes are present in 15% of Alexa page requests. Finally, to understand the effects of flow contention, we experimented with SPDY and found that depending on the server locations, SPDY can reduce some page load times in Ghana by up to 30%.

IMC'14, November 5–7, 2014, Vancouver, BC, Canada.

ACM 978-1-4503-3213-2/14/11 ...$15.00.
http://dx.doi.org/10.1145/2663716.2663748.

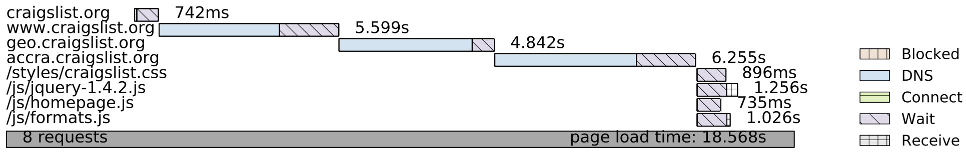


Figure 1: HAR trace visualization of `craigslist.com` from Accra, Ghana cellular in 2012

2. MOTIVATION

Information and Communication Technologies (ICTs) have been linked to development [26], where in the last four years, developing nations have experienced a significant increase in the number of Internet users, which is a promising trend. Between 2009 and 2011, Ghana experienced a huge increase in the number of Internet users from 5.4% to 14.1% [12], which is the largest increase in the country's history. A similar trend is being reflected in other developing countries, resulting in millions of new Internet users every year. However, the Internet that these users encounter is often extremely slow and cannot take advantage of all available content on the web.

Very little effort has been expended on understanding and solving the Internet problems in developing regions relative to the number of people these issues affect. Even very basic questions like why the Internet is so slow is a challenging question to answer due not only to the myriad of difficulties in collecting data in these regions, but also due to the multitude of answers depending on the context. Interestingly, when these problems are investigated, findings range widely from the relatively obvious, e.g. low bandwidth, high latency, unreliability, and loss, to the esoteric, e.g. mis-routing due to lack of peering points, small-packet regimes, etc. [20, 13, 25, 16]. Solutions to these problems are similarly varied [24, 23, 17]. Motivated by these differing ideas, in this paper we seek to answer why page loading times are so slow in Ghana.

3. METHODOLOGY

We used standard off-the-shelf tools to collect our traces. We collected HTTP Archive (HAR) browser-level traces using Firefox Version 13 with the Firebug extension and Selenium [7] to automate the process. The browser cache is cleared after each webpage. We also used the Linux dig command-line tool to examine DNS requests in more detail.

Table 1: Experiments details

Time	Type	Link	Location
August 2012	Alexa's top 500 global	Both Wifi & 3G	Accra and Hohoe (3G)
March - July 2013	DNS dig+trace	Wifi	Accra, Abu Dhabi NY and Bremen
August 2014	DNS dig+trace	Wifi	Accra
April 2014	SPDY	Wifi	Accra, Abu Dhabi NY and Bremen
August 2014	Alexa's top 1000 global	Both Wifi & 3G	Accra, Abu Dhabi NY, Bremen and Kumawu (3G)
August 2014	Alexa's top 500 local	Both Wifi & 3G	Accra, Abu Dhabi NY, Bremen and Kumawu (3G)

We ran our experiments from six locations in total: New York City, Bremen (Germany), Abu Dhabi (UAE), Accra (Ghana), Kumawu and Hohoe (both small cities in Ghana). In New York, Bremen, Abu Dhabi, and Accra the machines we used were state-of-the-art with a standard Linux distribution (Ubuntu and CentOS), and the experiments were conducted on university campus networks with over 10Mbps connections using Wifi connectivity and 3G HSPA cellular network connections. In Hohoe/Kumawu, our traces were collected from a 3G HSPA connection. In Hohoe, the fastest wired network was limited to ADSL at a single Internet Cafe and we were not able to collect data from that location. For our experiments, we collected web traces of Alexa's [2] top 1000 global and top 500 local websites, collected results of DNS performance using the "dig +trace" command, and evaluated SPDY performance in Ghana, see Table 1.

4. DISSECTING PAGE LOADS

In this section we focus on identifying problems associated with the webpage download process in Ghana by examining the set of browser-level traces we collected from Ghana in 2012.[1] Some of the timing information we are able to collect from the HAR files include:

- **Blocking**: waiting time of a request in the queue, due to the maximum number of parallel TCP connections that a browser can open per server
- **DNS Lookup**: time required to resolve a host name or an URL
- **Connecting**: TCP connection setup time
- **Sending**: the time required to send HTTP request to the server
- **Waiting**: the time required to receive a response from the server
- **Receiving**: the time required to read the entire response from the server

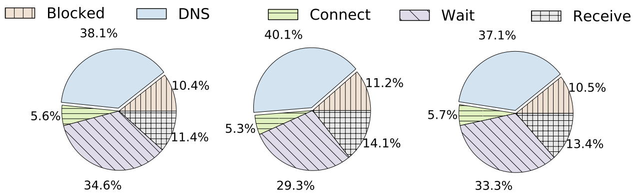
Figure 1 gives an example of a HAR file visualized as a waterfall chart.

4.1 Domain Name Server (DNS)

From our Ghana traces, the immediate observation is that the dominant contributor to the long page download times is the DNS lookup. Figure 2 shows the statistics taken from our measurements in Ghana 2012. The timing breakdown categories (i.e., "Blocking" and "Waiting") are identical to those defined previously in Section 4. We observe that DNS requests account for 37-40% of the user-perceived latency.[2]

[1]While the issues we observe here serve to highlight the extreme nature of the latency problem in developing countries, Accra actually has one of the fastest Internet infrastructures in Africa. Other developing regions suffer from even worse network conditions, e.g. South Africa has higher latencies [30], East Africa and Southeast Asia have fewer major network cables [10].

[2]A number of these requests occur simultaneously and we removed such overlapping timings. However, these figures do still include timings that do not contribute to the overall page load time, because they are not on the critical path.



(a) Accra 3G (b) Accra Wifi (c) Hohoe 3G

Figure 2: Web page requests (Alexa's top global 2012)

Although, "Wait" accounts for roughly 30% of the load time in all cases, we do not focus on it because link latency challenges in developing countries like Ghana have already been studied elsewhere [18, 25]. In these studies, the authors suggested several methods to improve the link latencies by investing in local server infrastructure or placing web caches and content on servers within the local service provider.

4.2 HTTP redirects

Looking more closely at the Ghana traces, we find that in about 80% of the cases, the first HTTP GET request actually gets redirected into another page. These redirects are caused either by the HTTP status code response of 301 Moved Permanently or 302 Found. We find that in most cases a redirection is due to directing a browser to a www URL of the same page, i.e. if we look at the example shown in Figure 1, we can see that the client asks for page `craigslist.org` and gets redirected to the domain `www.craigslist.org`.

4.3 HTTP blocking

Another interesting effect that we observe is blocking. In Figure 2 we see that blocking causes more than 10% of the page load time. This is due to the configured maximum number of TCP connections to a server within the web browser. From the example in Figure 3, we see that the browser has already requested six objects simultaneously using six TCP connections (style_splash.css, logo.jpg, etc.), and the next object that has to be requested from the same server (header_left.jpg) now has to wait until one of the previous object is downloaded.

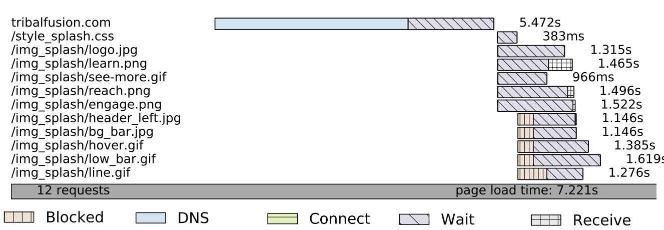


Figure 3: Blocking example of `tribalfusion.com` from Accra, Ghana cellular in 2012

In addition to the maximum number of TCP connections, modern web browsers also use HTTP pipelining with $4-8$ pipelined objects. HTTP pipelining is a way for the client browser to request several objects at the same time within one TCP connection, but Head-Of-Line blocking (HOL blocking) may occur because of this limit as well.

4.4 TLS/SSL

Our 2012 traces from Ghana showed that approx. 8% of the overall object requests require a secure connection. In 2014 this number has nearly doubled to 15%. This corroborates with the trend of increased adoption of secure content exchange on the web. However, in developing regions like Ghana this can severely impact user-perceived latency because establishing a Transport Layer Security (TLS) connection is a lengthy procedure and is severely impacted by latency. The TLS handshake procedure depends on whether only the server is authenticated or both the server and the client are authenticated, but the basic TLS handshake consists of up to 9 round trips [9].

5. SPEEDING UP PAGE LOADS

Here we assess the potential of several straightforward improvements that can cost orders of magnitude less than extending the existing network infrastructure.

5.1 DNS Caching

In order to understand the effect and potential gain of DNS caching, we analyze our traces in more detail. Figure 4 shows an example on long DNS requests of one of the webpages taken from the Accra 2012 cellular traces.

In this example, the page requests objects from *eleven* external domains, where a DNS request must be issued for each single domain. In order to evaluate the ratio of the DNS request to the total page time we calculate the overall delay caused by DNS. This is not a simple summation of all DNS requests, as can be seen in Figure 4. To calculate the potential benefit of DNS caching we leverage the Wprof tool [31] to identify the critical path in the page request and calculate the contribution to that critical path by DNS requests. We find that the ratio of the overall DNS delays to the overall page load time is about 72% of the overall page load time in Figure 4.

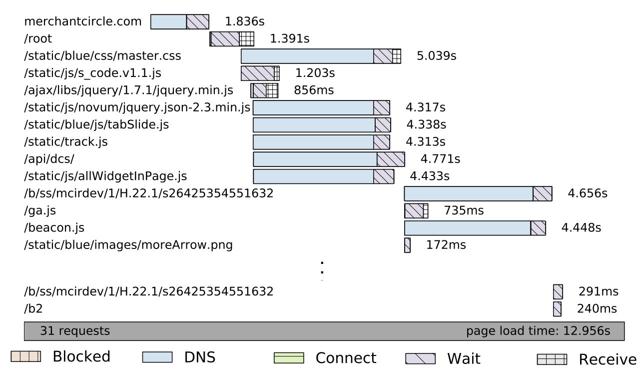
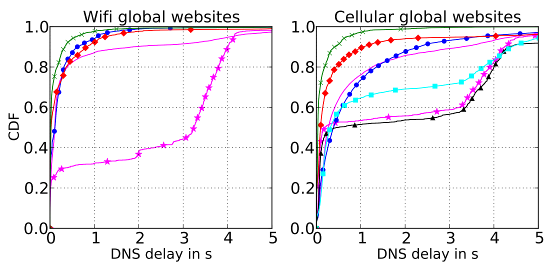


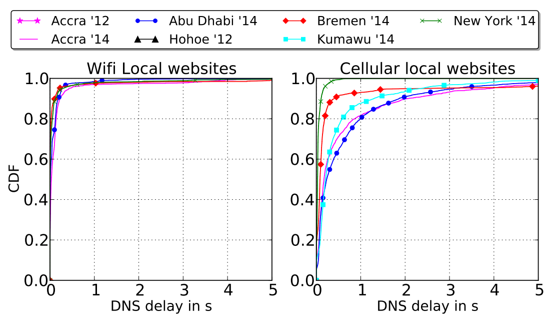
Figure 4: DNS example (`www.merchantcircle.com`) from Accra, Ghana cellular 2012

Figure 5 shows the DNS delays in our traces from Ghana, taken from all embedded domains within Alexa's top websites. The DNS delays are measured at the application layer by means of HAR files, thus they represent the user-perceived performance. These DNS delays also include all DNS retry attempts and redirects. The figure shows how the overall DNS performance in Ghana has changed over the past two years.

In 2012, Ghana's DNS CDFs (Figure 5a Wifi and cellular) split into two regions: a low delay region in the order of milliseconds and a high delay region in the order of seconds. When we examined the websites that belonged to the high delay group, we did not find any distinctive patterns, e.g., all are .com domains etc. However, by cross-referencing the domains behind each region, we found that 15% of the low delay domains were also present in the high delay region. This suggested that the low delay region is likely due to DNS caching at the local ISP DNS server.



(a) Alexa's top *global* websites

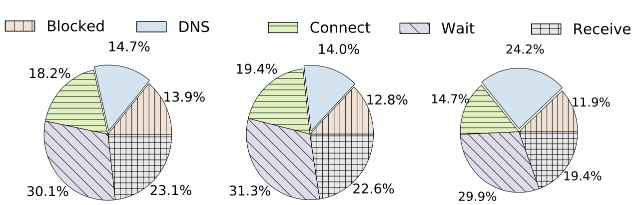


(b) Alexa's top *local* websites

Figure 5: Per location DNS delay (HAR files)

In 2014, we have redone new experiments in Ghana using Alexa's top 1000 global websites and top 500 local websites. We also compare the Ghana results to measurements performed in other locations around the world. We observe a clear improvement in Accra's Wifi DNS performance due to better DNS caching on the university network for both the global and the local measurements. However, the DNS performance over cellular is only improved in Accra's case, but Kumawu's cellular DNS performance is still very poor (Figure 5a right side).

Figure 6 shows the pie chart of the page load time in Ghana for the 2014 measurements. We can see that the DNS response times have improved compared to the 2012 measurements (Figure 2). In contrast, the TCP connection time has increased to 3-4 times in terms of proportion of overall page load time.



(a) Accra 3G (b) Accra Wifi (c) Kumawu 3G

Figure 6: Web page requests (Alexa's top global 2014)

5.2 DNS Server Placement

Besides caching, DNS server placement plays an important role to reduce DNS latency. There are over 300 root servers scattered around the globe operated by 12 different organizations. Although only 13 root servers appear in response to a request, these servers are reachable with 13 anycast IP addresses [6]. The root DNS servers are not provisioned uniformly around the world or in proportion to population densities. The authoritative DNS servers for Top-level domains (TLDs) is similarly unevenly distributed.

To get a better understanding of the high DNS delay region from Figure 5a, we analyzed the DNS performance using the dig command-line tool. The dig tool is used in Linux to query DNS nameservers. It can be combined with the "+trace" option to follow the full lookup process from the root servers up until getting the IP address of the requested domain while displaying the answer for each involved server.

The "dig +trace" output can be divided into different hierarchical categories starting with the first category where the local DNS server gives back the answer to the root servers (i.e., the '.' domains) all the way to the last category that gives back the IP address of the requested domain. Figure 7 shows the "dig +trace" results for Accra (2013 and 2014) in comparison with New York, Bremen, Abu Dhabi (2013). The total DNS delay depicted in Figure 7 is a combination of several hierarchical requests from the local DNS server, root server, TLD servers, and sub domains servers. Although, the number of categories of the DNS hierarchy can be more than four, we have observed that the most dominant number is four, which is why we only present the first four results. Figures 8a to 8d are CDFs of DNS delays per hierarchy.

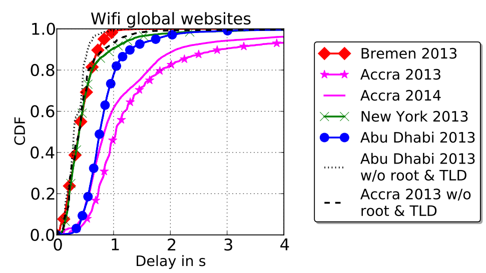
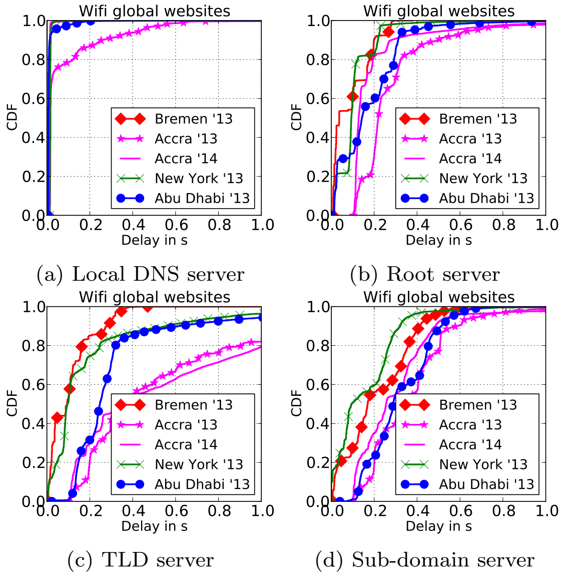


Figure 7: Per location DNS delay on Wifi ("dig +trace")

Figure 8a shows the local DNS server delay (to retrieve the root authoritative servers). Apart from Accra's 2013 upper 30%, the results show that the local DNS server replies back with an answer within milliseconds. For Accra's 2013 upper 30% some network performance issues degrade the performance. Figure 8b and 8c show the DNS delay of the root and TLD servers respectively. The results show that both New York's and Bremen's root and TLD authoritative servers have faster response times compared to Abu Dhabi and Accra. This indicates that these servers are geographically closer to New York and Bremen. Accra has the worst delay profile compared to all other locations. Figure 8d shows the delay CDF for the next DNS hierarchy. The result shows that Abu Dhabi and Accra have slightly higher delays than New York and Bremen. This is due to the increased geographical distance to the DNS resolvers. Although, in the previous section, we have seen better DNS performance for Accra's 2014 over Wifi network; the results depicted in this section show that the physical distance to the DNS resolvers is still a problem.



(a) Local DNS server
(b) Root server
(c) TLD server
(d) Sub-domain server

Figure 8: Per location/hierarchy delay CDFs

In order to evaluate the distance from Ghana to the DNS resolvers, we have performed several experiments using the destination, showing how many hops are in between, and the IP addresses of these hops. We used traceroute against all 13 root domain servers from Accra, recorded the routes, and repeated each experiment about 50 times. We found that nearly all of the traceroutes' first hops were outside Ghana and somewhere in Europe (Switzerland, UK, etc.), from where, the route sometimes diverts to the US or Asia. None of the routes went to DNS root servers that are geographically nearby.

In summary, there are large potential gains for enhancing the overall DNS delay in Ghana in case more root and TLD servers are deployed in that particular region. Figure 7 shows the potential performance gain when root and TLD servers are placed hypothetically with zero latency to Abu Dhabi and Accra. These additional root servers would give very similar performance to what is currently observed for New York and Bremen.

5.3 Caching Redirects

In section 4.2, we discussed the issue of the first HTTP GET redirection. One way to overcome this is to cache redirects, where the client could be informed about the redirection immediately. Caching redirection can enhance the overall page load time by about 20% (see Table 2).

Caching redirects can cause revocation problems since even after removing the redirection, the cached redirections cannot be removed and clients will still be redirected and never reach the new server.

Table 2: Redirects delay to total page load time (2012)

Location	Websites with Redirects	Average Ratio	Standard Deviation
Accra cellular	79%	25%	15%
Accra Wifi	78%	20%	16%
Hohoe cellular	81%	21%	17%

5.4 SPDY vs TCP Optimizations

SPDY [8] is an application layer protocol proposed to enhance the webpage loading time. The SPDY protocol includes header compression and allows servers to send back additional responses without the client asking for them, e.g., the server might send back the style sheet of a page along with the HTML, which can speed up the page load time.

Since blocking was an issue we identified in Section 4.3, we wanted to examine the utility of SPDY, as it offers the potential to reduce the overhead of blocking times. We requested 42 pages from four different locations: Accra, Abu Dhabi, Bremen, and New York.[3] Each experiment was repeated five times to have enough randomness for a better CDF population. Figure 9 shows the overall comparison between SPDY and HTTP for all locations.

For the 42 websites that supported SPDY in our experiments, we have found that SPDY does not show significant performance gain compared to regular HTTP in developed regions (e.g., New York, Bremen). Whereas, in Accra, SPDY shows a performance gain in the upper 40% of the page load times. This is because SPDY reduces the number of TCP connections by multiplexing HTTP requests into a single TCP connection per domain. As found in [32], SPDY reduces the page load times especially for links with less bandwidth and higher RTTs.

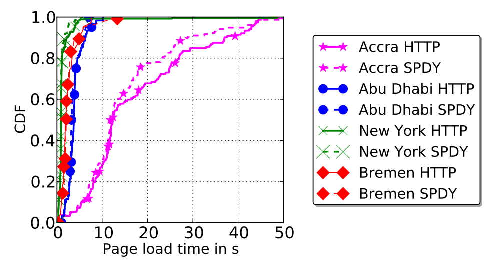


Figure 9: Per location SPDY vs HTTP in 2013

In Ghana where the connections are latency, not bandwidth, constrained (at least for the purposes of web browsing), SDPY is a better solution than multiple TCP connections. Almost all popular browsers are configured by default to use multiple persistent TCP connections when downloading a page over HTTP to avoid HOL blocking, but increasing the number of persistent TCP connections does not always improve the HTTP performance [27]. Furthermore, in developing countries where the bandwidth is scarce and sharing is high, increasing the number of TCP sessions can degrade HTTP performance [16].

5.5 TLS/SSL Speedups

Although the TLS protocol has a resume handshake that significantly reduces the handshake traffic and latency, it requires the server to store the TLS session in its cache. Thus, many servers store the TLS session only for a short period of time resulting in the full handshake being used instead. Approximately 15% of our latest web requests contained a secure connection establishment handshake. Improving handshake procedure can reduce the overall load time, and combining the handshake with SPDY can potentially im-

[3]These were the only websites out of the Alexa's top 500 websites that supported SPDY.

prove the overall page load time without adding significant computational load on the servers [3].

Fast-Track [28] is a proposed mechanism where the server's public and negotiated parameters are cached at the client side, since the server's parameters are nearly static. This helps reduce the overall TLS handshake traffic and the number of RTTs required, thus providing substantial gain in high latency developing countries. Unfortunately, Fast-Track requires modifications to both the client and the server side by means of TLS extensions and not all servers will have the Fast-Track support option implemented. We do not present measurements of the potential of TLS/SSL speedups of these solutions because they require control over the server.

6. OTHER RELATED WORK

There is a wide range of related work in the networking literature and we include in this section only works we have not yet referenced. Stream Control Transmission Protocol (SCTP) is a transport-layer protocol to replace TCP, which provides multiplexed streams and stream-aware congestion control [29]. HTTP 1.1 over SCTP may be used to avoid the HOL blocking problem where when one transport protocol data unit (TPDU) is lost, TCP does not deliver successive TPDUs until the one that is lost is recovered. Structured Stream Transport (SST) is a protocol that uses "structured streams": lightweight, independent streams to be carried over a common transport sharing the same congestion control context [22].

Recent work [32] investigates the conditions under which SPDY provides improvements over HTTP. The work shows that SPDY performs better when few network losses, high RTTs and low bandwidth are given. We corroborate these findings for high RTT conditions in developing countries like Ghana. In [21], the authors propose a simple solution to increase TCP's initial window size to at least ten segments, which may also help for developing regions.

7. CONCLUSIONS

In this paper, we have presented an analysis of the key factors that trigger high end-to-end latency for web page downloads in Ghana. From our measurements over the past two years, we observed that DNS is a critical bottleneck in page load times and even the deployment of simple ideas such as DNS caching can help improve the end-to-end performance. Apart from DNS, we found that because of the lack of server infrastructure within the country, every request needs to be routed to US or European countries thereby incurring a high network latency penalty; hence, even simple operations such as HTTP redirections and TLS/SSL handshakes represent a non-trivial fraction of the page load time. We experimented with a host of well known optimizations and describe the effectiveness of some of the promising approaches including DNS caching, DNS server placement, caching redirects, using SPDY, and TLS/SSL speedups.

8. ACKNOWLEDGEMENTS

We thank Kessir Adjaho and David Hutchful for their help in collecting measurement data from Ghana. We would also like to thank our shepherd Harsha V. Madhyastha and the anonymous reviewers for their valuable feedback. Talal Ahmad was partially supported by the Center for Technology and Economic Development at NYUAD (CTED).

9. REFERENCES

[1] Akamai: State of the Internet. `http://www.akamai.com/stateoftheinternet`. Accessed: 2014-05-04.

[2] Alexa - The top 500 sites on the web. `http://www.alexa.com/topsites`. Accessed: 2014-04-05.

[3] Is TLS Fast Yet? `https://istlsfastyet.com/`. Accessed: 2014-08-15.

[4] Make the Web Faster - Web metrics: Size and number of resources. `http://developers.google.com/speed/articles/web-metrics`. Accessed: 2014-05-04.

[5] Ookla Download Index 2012. `http://www.netindex.com/download/allcountries/`. Accessed: 2013-02-05.

[6] Root Servers. `http://www.root-servers.org/`. Accessed: 2013-03-15.

[7] SeleniumHQ Browser Automation, Version 2.41.0. `http://docs.seleniumhq.org/`.

[8] SPDY: An experimental protocol for a faster web. `http://www.chromium.org/spdy/spdy-whitepaper`. Accessed: 2014-05-04.

[9] SSL Profiles: Part 1 by Jason Rahm. `http://devcentral.f5.com/articles/ssl-profiles-part-1`. Accessed: 2014-04-05.

[10] The World in 2010: ICT Facts and Figures - ITU. `http://www.itu.int/ITU-D/ict/material/FactsFigures2010.pdf`. Accessed: 2014-05-04.

[11] Web Site Optimization: Maximum Website Performance. `http://www.websiteoptimization.com/`. Accessed: 2013-05-04.

[12] World Bank Group - World Development Indicators. `http://data.worldbank.org/country/ghana`. Accessed: 2013-04-30.

[13] E. Brewer, M. Demmer, M. Ho, et al. The challenges of technology research for developing regions. *Pervasive Computing, IEEE*, 5(2):15–23, 2006.

[14] M. Butkiewicz, Z. Wu, S. Li, et al. Enabling the transition to the mobile web with websieve. In *Proc. of the 14th Workshop on Mobile Computing Systems and Applications*, HotMobile, pages 14:1–14:6, New York, NY, USA, 2013.

[15] J. Charzinski. Traffic properties, client side cachability and cdn usage of popular web sites. In B. Müller-Clostermann, K. Echtle, and E. P. Rathgeb, editors, *MMB/DFT*, Lecture Notes in Computer Science, pages 136–150. Springer.

[16] J. Chen, L. Subramanian, J. Iyengar, and B. Ford. TAQ: enhancing fairness and performance predictability in small packet regimes. In *Proc. of the 9th European Conference on Computer Systems*, page 7. ACM, 2014.

[17] J. Chen, L. Subramanian, and J. Li. Ruralcafe: Web search in the rural developing world. In *Proc. of the 18th International World Wide Web Conference*, 2009.

[18] M. Chetty, S. Sundaresan, S. Muckaden, et al. Measuring Broadband Performance in South Africa. In *Proc. of the 4th Annual Symposium on Computing for Development*, pages 1:1–1:10, New York, NY, USA, 2013. ACM.

[19] Sahuquillo J. Pont A. Domenech J., Gil J.A. Ddg: An efficient prefetching algorithm for current web generation. In *Proc. of the 1st IEEE Workshop on Hot Topics in Web Systems and Technologies (HotWeb)*. IEEE, 2006.

[20] B. Du, M. Demmer, and E. Brewer. Analysis of WWW traffic in Cambodia and Ghana. In *Proc. of the 15th International Conference on WWW*, pages 771–780, 2006.

[21] N. Dukkipati, T. Refice, Y. Cheng, et al. An Argument for Increasing TCP's Initial Congestion Window. *SIGCOMM Comput. Commun. Rev.*, 40(3):26–33, June 2010.

[22] B. Ford. Structured streams: A new transport abstraction. In *Proc. of the Conference on Applications, Technologies, Architectures, and Protocols for Computer Communications*, SIGCOMM '07, pages 361–372, New York, NY, USA, 2007. ACM.

[23] S. Guo, MH Falaki, EA Oliver, et al. Very low-cost internet access using KioskNet. *SIGCOMM Comput. Commun. Rev.*, 2007.

[24] B. Hull, V. Bychkovsky, Y. Zhang, et al. CarTel: A distributed mobile sensor computing system. *Sensys*, pages 125–138, 2006.

[25] Z. Koradia, G. Mannava, A. Raman, et al. First impressions on the state of cellular data connectivity in india. In *Proc. of the 4th Annual Symposium on Computing for Development*, Cape Town, South Africa, 2013.

[26] O. Ngwenyama, F. K. Andoh-Baidoo, F. Bollou, and O. Morawczynski. Is There A Relationship Between ICT, Health, Education And Development? An Empirical Analysis of five West African Countries from 1997-2003. *The Electronic Journal of Information Systems in Developing Countries*, 23, 2006.

[27] F. Baker P. Natarajan and P. D. Amer. Multiple TCP Connections Improve HTTP Throughput Myth or Fact? In *Proc. of the IEEE 28th Int'l Performance Computing and Communications Conference*. IEEE, 2009.

[28] H. Shacham and D. Boneh. Fast-track session establishment for TLS. In Mahesh Tripunitara, editor, *Proc. of NDSS 2002*, pages 195–202. Internet Society (ISOC), 2002.

[29] R. Stewart. Stream Control Transmission Protocol, September 2007. RFC4960.

[30] S. Sundaresan, N. Feamster, R. Teixeira, and N. Magharei. Measuring and Mitigating Web Performance Bottlenecks in Broadband Access Networks. In *Proc. of the ACM Internet Measurement Conference*, Barcelona, Spain, 2013.

[31] X. S. Wang, A. Balasubramanian, et al. Demystify Page Load Performance with WProf. In *Proc. of the USENIX conference on Networked Systems Design and Implementation (NSDI)*, 2013.

[32] X. S. Wang, A. Balasubramanian, A. Krishnamurthy, and D. Wetherall. How Speedy is SPDY? In *Proc. of the 11th USENIX Symposium on Networked Systems Design and Implementation (NSDI)*, pages 387–399, Seattle, WA, 2014.