Byte-By-Night: Web Cache Object Forwarding From Desktop to Mobile for Energy Consumption Optimizations

Troy Johnson
Department of Computer Science
Central Michigan University
Mount Pleasant, MI 48859
johns4ta@cmich.edu

Patrick Seeling
Department of Computer Science
Central Michigan University
Mount Pleasant, MI 48859
pseeling@ieee.org

***Abstract*—**

***Index Terms*—Mobile communication; Middleware; Energy consumption**

I. Introduction

With the beginning of the 21st century, new support for mobile connected user devices has fueled a continuous increase in the demand for mobile data. Web requests now originate in a majority from wirelessly connected user devices, a trend that Cisco, Inc. predicts to continue in the foreseeable future [1]. Simultaneously, the overall user behavior and demand for more rich media inclusion into web pages has increased the overall amount of data that is required to be transmitted per page, see, e.g., [2], [3]. Caching on the client side has been effectively used in the past and was, together with increased numbers of parallel object downloads, able to decrease wait times for desktop clients as reported in [3]. A first view of mobile web page characteristics, which were found to exhibit lower complexity, and non-landing pages, which were found to be less complex than landing pages, was given by the authors of [2]. Moreover, a significant body of research has emerged that focuses on content delivery optimizations to mobile devices.

Typically, these optimization approaches are typically targeting on-device optimizations or off-device cloud-based optimizations. For mobile applications, for example, significant energy savings were found to be attainable when grouping application requests so as to avoid prolonged cellular network interface activity, see, e.g., [4], [5]. Other approaches optimize the delivery to mobile devices through proxies and cloud-based data anticipation and traffic shaping, see, e.g., [6].

For web data, typically caching is used to limit the amount of data that has to be transfered to requesting clients. In prior works focusing on mobile web page delivery optimizations, such as [7], energy optimization has been a key element, due to the restrictions for mobile clients. One particular area for optimization is pre-fetching combined with caching, which allows to limit downloads, such as presented with about 10 % energy savings in [8]. More recently, these approaches were combined with user connectivity predictions, as in, e.g., [9].

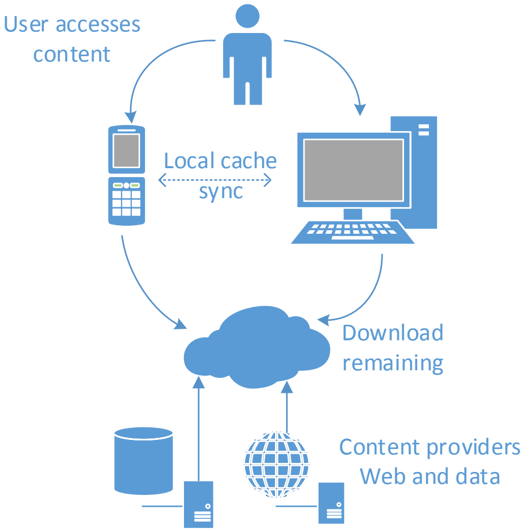


Fig. 1. Overall setup of world wide web based content access by a user over time using primary and secondary devices to display the same content. If the devices are able to synchronize their caches through short-range communications, energy savings become possible.

We propose to utilize a basic cache forwarding method that can be utilized by users to synchronize from, e.g., a desktop computer, to their mobile device, e.g., a smartphone. We illustrate this approach in Figure 1, which contains the desktop and a mobile device. As most devices are charged over night, or are at least stationary within local area network ranges, we propose to utilize this general idle period to allow direct forwarding of cached objects that are identical between the different device display modalities, i.e., identical web objects delivered when requesting a desktop/mobile versions of a web page. The thus transmitted objects now reside in the local device cache and do not require an energy-expensive mobile download through cellular interfaces. To support our approach, we gather data through the publicly accessible WebpageSpeedTest.org website as well as the httparchive.org

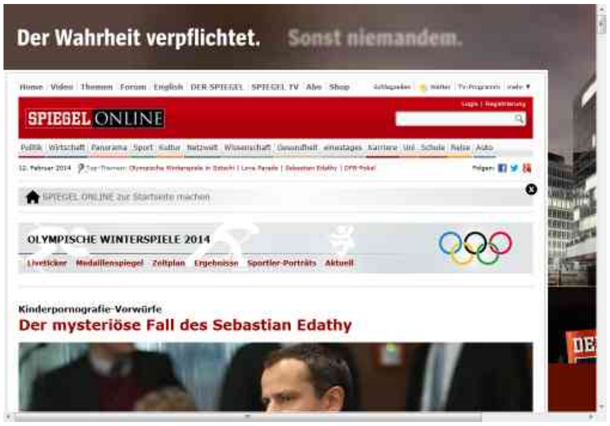

Der Wahrheit verpflichtet. Sonst niemandem.
SPIEGEL ONLINE
SPIEGEL ONLINE zur Startseite machen
OLYMPISCHE WINTERSPIELE 2014
Liveticker Medaillenspiegel Zeitplan Ergebnisse Sportler-Porträts Aktuell
Kinderpornografie-Vorwürfe
Der mysteriöse Fall des Sebastian Edathy

Fig. 2. Screenshot of rendered web page from WebPageSpeedTest.org. As illustrated, the main textual and pictorial content items are flanked by background and interactive advertisements.

archive of a large dataset of performance evaluations for popular web pages; we refer the interested reader to [10] for a more detailed discussion.

The remainder of this paper is structured as follows. In the following section, we provide an example based on a particular news web page. In the subsequent Section III, we discuss the overall properties of a large web page dataset archive and provide a description for the simulation-driven evaluation of our approach. We discuss the obtained results in Section IV before concluding in Section **??**.

II. Individual Example

In this section, we outline an individual example for the German news web page for "Der Spiegel," accessible at http://www.spiegel.de as an example for a rich media and advertising material containing web presence that is frequently updated. On [insert final date], we performed a speed test online using Internet Explorer as browser instance for a desktop client, Chrome as mobile browser on a Google Nexus 5 instance, and Safari for an iPhone 4 instance provided by WebpageSpeedTest.org (both mobile clients were traffic shaped to 3G connection emulations). An example screenshot of the web page as rendered is provided in Figure 2. As illustrated, the web page exhibits a significant amount of objects that are required for advertisements, visual items (images, layout), and scripting. We evaluate the reported data as follows. For each display modality and browser client, we gather the individual web objects requested and the response sizes and headers for caching information (for HTTP response codes 200 only, as we are not interested in redirects). We denote the number of all objects returned for a request modality (i.e., Internet Explorer for the desktop client example and iPhone or Nexus 5 for mobile counterparts) as N, their average size as $\overline{X}$, the standard deviation of their sizes as σ_X, and the Coefficient of Variation (CoV) of the returned object sizes as CoV_X. As in the HTTP specification, the *max-age* directive overrides others, so we initially consider that directive for the cache longevity of the individual objects. If no explicit information is found, we consider the header's *expires* information, which provides a secondary cache lifetime. If both are not found, or if the *expires* date is in the past or right at the request time, we set the cache lifetime we consider here to zero. Similar to the notation for the object sizes, we denote the expiration time characteristics as $\overline{T}$, σ_T, and CoV_T, respectively. In case of size differences between the three, we utilize the smallest size as lower boundary.

TABLE I
High-level overview of different client webpage statistics for http://www.spiegel.de

Statistic	IE Deskt.	iPhone	Nexus 5
N	154	132	145
$\overline{X}$	10402.77	8846.39	10201.63
σ	19103.19	13438.05	25684.58
CoV_X	1.84	1.52	2.52
$\overline{T}$	106285.14	114780.69	114818.28
σ_T	125080.27	125576.65	126090.90
CoV_T	1.18	1.09	1.10

A. Data Description

We provide an initial high-level overview of the webpage characteristics for http://www.spiegel.de as requested in Table I.

We initially observe that the desktop version (with Internet Explorer 9 as requesting browser client) exhibits the highest number of objects, followed by the mobile Chrome/Android and Safari mobile/iOS versions, respectively. Interestingly, there is only a minor difference in the number of objects between these versions. Next, we observe that the trend for the average number of bytes is aligned with the number of elements. In total, the mobile Chrome version is almost on par with the desktop one (at 92%) and only the Safari mobile version seeming optimized (at 72%). The standard deviation and Coefficient of Variation (CoV) amongst individual element sizes are highest for the mobile Chrome version, indicating more significant size differences than for the safari version, which is one unit lower, while the desktop version falls into the middle.

Next, we evaluate the elements' caching properties on a high level in Table I. Overall, we note that the highest average caching time is observed for the mobile Chrome access with the Nexus 5 device, trailed immediately by the iOS access and with distance by the desktop browser. We note that the overall variability in terms of expiration times amongst elements is fairly low and comparable, as indicated by CoV values around 1.1.

B. Evaluation of Local Cache Forwarding

We now shift the view to the possibility for identical objects to be re-utilized locally to avoid additional download penalties in time and energy consumption. Simultaneous with a web page request, a local broadcast could "ask" for the cached

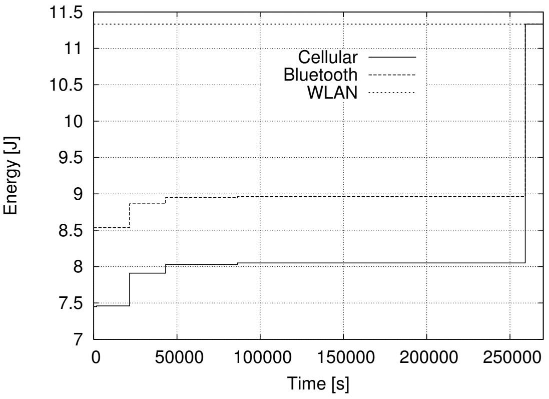


Fig. 3. Energy required to download web page data via cellular connection or through combination with partial local exchange with Bluetooth/WLAN.

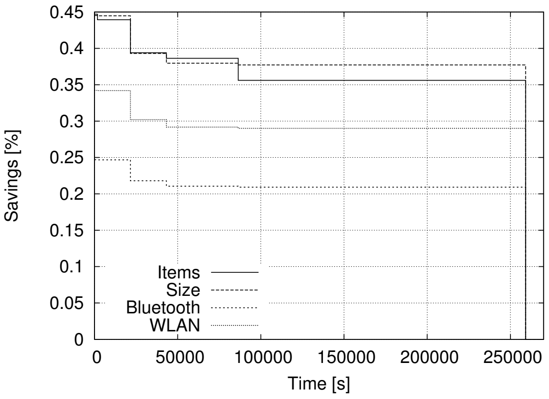


Fig. 4. Relative number of items and data amounts in cache and savings resulting from local data exchange with a desktop client.

elements for a website to be delivered locally form participating clients. Alternatively, cellular provisioning methods, such as in LTE-A, could initiate the device-to-device local exchange as well. Once the request is received, the local coordination can take place, which inherently results in the forwarding of elements that are the same across browser instances and which have a future cache lifetime expiration.

We filter out elements that are not similar amongst the different requesting devices, which results in 82 objects with the same URL and size combination. While we note an identical cache lifetime for most of these, a slight increase is notable from the Desktop over Safari to Chrome. Next, we remove items that exhibit a cache lifetime of zero for any of the three requesting clients, resulting in 59 objects, which now exhibit a reduced average cache lifetime for the mobile browsers, whereby mobile Chrome exhibits the shortest. In turn, we choose the iOS Safari mobile as our exemplary base for calculations. We utilize the approximations for energy consumptions presented in [11] to determine the amount of energy required as time progresses if local data exchange can be performed using either Bluetooth or WLAN technologies. As illustrated, the complete download energy for cellular data is the upper limit on energy spent downloading the data associated with the web page. The Bluetooth and WLAN data exchanges with other local clients both follow the same underlying trend with increasing energy required to transmit all data as time progresses from the last access point in time.

We illustrate the relative amount of data/items within the cache in addition to the attainable savings in Figure 4 as a function of delayed access time. We observe that a significant amount of the overall data can be cached (and would in turn be suitable for localized sharing).We additionally observe that even for longer durations up to multiple days, significant amounts of data remain in the cache. Combining the streaming from locally cooperating clients, e.g., user-owned desktop computer in the same network through either Bluetooth or WLAN technologies, we utilize the approximations outlined in [] to derive the relative gains of our approach in terms of energy consumption. Specifically, we compare our approach (combining locally exchanged caches and cellular downloads of the remainder) with the regular download of all items through the cellular network interface. We note that significant savings above 20 % are attainable with our approach, even if the filling of the device's cache is performed wirelessly as well.

III. Large Dataset Performance Evaluation

We now evaluate our proposed approach through simulations based on a large dataset of landing web pages of the most popular websites. For this purpose, we retrieved the publicly available web page response details from *httparchive.org* and create a subsequent dataset for the October 1st, 2013 dataset.

A. Dataset Description

The dataset we utilize for the performance evaluation contains the individual web responses for the most popular websites of fixed (Internet Explorer for the desktop) and mobile (iOS for iPhone 4) websites, ranked through the Alexa popularity index. Furthermore, the dataset contains the individual response details, including cache expiration times and sizes.

Similar to the individual web page example, we provide an overall description of the response sizes and maximum expiration ages in Table II. We note that the overall average response size as result of desktop requests (when batched into the outlined time frames) is around of 17 kB, with the largest average sizes occurring between one and twelve hours. Mostly, we note that the majority of objects exhibit a short expiration time between zero and 30 seconds. For the responses to mobile devices, we note that the majority of objects are in the same short time span and the long time span of more than one

TABLE II

OVERVIEW OF THE LARGE DATASET CHARACTERISTICS FOR ALL PAGES AND RESPONSE OBJECTS WITH A FOCUS ON THE CACHE LIFETIME.

Category	Desktop (IE)				Mobile (iOS)			
	$\sum X_i$ [MB]	N	$\sigma(X_i)$ [kB]	$\overline{X_i}$ [kB]	$\sum X_i$ [MB]	N	$\sigma(X_i)$ [kB]	$\overline{X_i}$ [kB]
expAge $<$ 30 Seconds	238490.28	15670679	1698.59	15.22	1355.24	122598	582.98	11.05
30 $\leq$ expAge $<$ 1 Minute	170.94	29130	27.38	5.87	12.68	898	70.38	14.12
1 Minute $\leq$ expAge $<$ 1 Hour	15192.40	797450	270.06	19.05	282.36	15854	226.62	17.81
1 Hour $\leq$ expAge $<$ 12 hours	29798.68	1186947	751.45	25.11	244.14	17424	209.84	14.01
12 hours $\leq$ expAge $<$ 1 day	6188.57	358998	227.63	17.24	91.37	5952	91.37	15.35
1 day $\leq$ expAge	195784.64	9888147	1337.36	19.80	2122.96	126629	750.51	16.77
Average	80937.59	4655225	718.75	17.39	684.79	48226	321.95	14.20

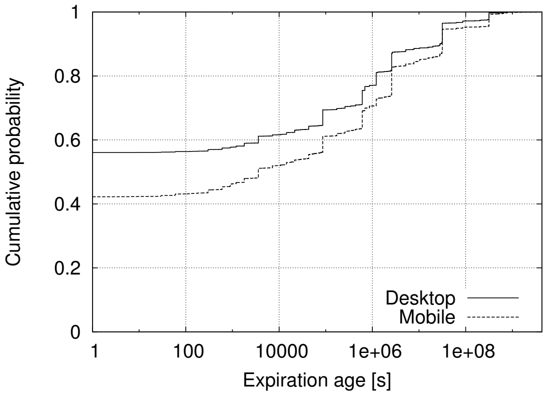


Fig. 5. Cumulative probability

day. The highest average response sizes, however, are observed between one hour and half a day, in difference to the desktop counterpart.

We illustrate the cumulative distribution for the desktop and mobile client response expiration ages in Figure 5. We observe that a significant portion ($>$ 40 %) of responses form web servers exhibits an immediate expiration and would, in turn, require an immediate retrieval by clients. The cumulative expiration probability is higher for responses to desktop clients, which can be attributed to the typically more abundant bandwidth (and less optimization requirements) in contrast to a mobile setting (where web content developers might be more considerate of limitations).

B. Simulation Model Description

We evaluate the effectiveness of our approach through simulation, which we perform as follows. For the time period of one week, we randomly select a web page from the mobile dataset and separate the data that is required to be downloaded and the data that is still available in cache due to the maximum expiration age of the individual objects. We subsequently wait for a random amount of time before repeating the procedure. We now describe the process in greater detail.

Initially, we sort all landing web pages $l, l = 0, \ldots, L$ from the Oct. 1 dataset in the *httparchive.org* dataset based on their web page popularity ranking. To simulate the popularity of web pages, it was found that the Zipf distribution effectively describes the popularity of individual requests, see, e.g., []. We chose to utilize the non-modified Zipf distribution for our simulation purposes, noting that additional modifications to the distribution exist that can further enhance it, see, e.g., [12]. For the total number of web pages L, we derive the probability of page l being selected in turn as

$$p(l) = \frac{l^{-\alpha}}{\zeta(l)}, \tag{1}$$

whereby $\zeta(\cdot)$ denotes the Zeta function. We furthermore set $\alpha = 0.85$ as an average choice in common ranges utilized for this parameter.

Once the individual mobile web page l was randomly chosen, we compare it to its desktop counterpart, with $\{d, m\}$ denoting the request modality. As each page l exhibits a time-sensitive number of responses, we denote them as $r^{\{d,m\}}(l, t)$ and each page exhibits $R^{\{d,m\}}(l, t = 0)$ in total. Let $t_c(r^{\{d,m\}}(l))$ denote the expiration or max-age directive received with the response at $r^{\{d,m\}}(l, t = 0)$. The number of objects or responses to be retrieved at time t in turn are given as

$$R^{\{d,m\}}(l, t) = \sum_{r^{\{d,m\}}(l,0)} \left[t_c(r^{\{d,m\}}(l)) \geq t \right] \tag{2}$$

where $[\cdot]$ denotes the Iverson Bracket. In the following, we abbreviate to t_c for readability when in direct context.

We furthermore denote the size of the individual response retrieved at time t as

$$x_r^{\{d,m\}}(l, t) = x_r^{\{d,m\}}(l, 0) \cdot [t_c \geq t] \, . \tag{3}$$

The total size of objects or responses retrieved at simulation time t thus is given as

$$X^{\{d,m\}}(l, t) = \sum_{r^{\{d,m\}}(l,0)} x_r^{\{d,m\}}(l, t). \tag{4}$$

For performance evaluation purposes, we simulate the user behavior similar to the process outlined in [13], whereby we randomly draw the time between requests t_u as Pareto distributed using $p(t_u) = \beta k^{\beta} t_u^{-(\beta+1)}$ with $\beta = 1.5$, $k = 30$. We simplifyingly assume that no delays are accrued due to instantaneous cache retrievals or downloads. Using time index i, we thus derive $t_{i+1} = t_i + t_u$. We continue the simulation until $t_{i+1} = t \geq T = 640800$ [s], i.e., one week.

IV. Performance Evaluation Results

We preset our results for the individual responses or web objects and the total number of bytes, noting that through approximations such as the ones described in Section II an inference of energy consumption would be possible as well. We perform the simulations with an average $\alpha = 0.85$ for the utilized Zipf distribution and repeat each week-long evaluation 2000 times. Initially, we present the total number of responses and bytes that would not require a download (i.e., would reside in the mobile cache) as a function of time in Figure 6.

We note an immediate decrease in the number of requests, which is mirrored by the umber of bytes as well (indicating a linear relationship as observed in prior works). We furthermore observe that the initial decline levels at the time of around one day, and then remains steady afterwards. This indicates that within the simulation period of one week, initially a large number of items is non-shared between devices or exhibits no cache lifetime. In a following group of objects, cache lifetimes increase logarithmically, which leads to the exponential decline we observe here. The narrow confidence intervals and low level of standard deviation indicate that the presented results are stable within simulation confines.

Next, we present how these findings translate into attainable savings with respect to requests for objects and bytes in Figure 7 in relationship to the total web page data without caching.

We initially note a declining trend similar to the one observed for the cache items in Figure 6, but with more distinct "jumps" in the data at half-day and day times of the simulation. Overall, we note that almost 15 % savings for objects and bytes, the savings decline to around 10 % after the boundaries of a day (with objects exhibiting lower, bytes higher levels).

Overall, we find a significant reduction in the required downloads for mobile clients that can be directly attributed to lower energy consumption levels and potential download latencies (and even costs for the mobile user through saved data transmissions). Furthermore, this approach does not require sophisticated live evaluations, but only relies on cache forwarding to the mobile device.

V. Conclusion

We exemplary evaluated the content of modern web pages and the identical objects that can be found when requesting the same web page from different devices with different display modalities, such as desktop and smartphone. Assuming mobile users visit the same pages on their mobile device that they visit on their desktop computer as well, we simulated the access of landing web pages and evaluated the potential for cache forwarding. Our basic approach is able to save upward of 7.5 % of mobile request or bytes even for very distant time horizons. In the more typical daily time range, our approach can yield almost linearly decreasing savings stating at about 14 %, without requiring any sophisticated prediction mechanisms, but only a direct cache transfer.

We note that similarly, exchanges between mobile devices belonging to a user are possible, including keeping a cloud-based reference of pages that a user visits (similar to Google Chrome's synchronization features), which is the subject of our ongoing work. We similarly note that incorporation of predictive caching and user context can with little overhead increase the outlined savings, which represents another current research venue.

References

[1] Cisco, Inc., "Cisco visual networking index: Global mobile data traffic forecast update, 2013–2018," Tech. Rep., feb 2014.

[2] M. Butkiewicz, H. Madhyastha, and V. Sekar, "Characterizing web page complexity and its impact," *Networking, IEEE/ACM Transactions on*, vol. PP, no. 99, pp. 1–1, 2013.

[3] S. Ihm and V. S. Pai, "Towards understanding modern web traffic," in *Proceedings of the 2011 ACM SIGCOMM conference on Internet measurement conference*, ser. IMC '11. New York, NY, USA: ACM, 2011, pp. 295–312. [Online]. Available: http://doi.acm.org/10.1145/2068816.2068845

[4] N. Balasubramanian, A. Balasubramanian, and A. Venkataramani, "Energy consumption in mobile phones: A measurement study and implications for network applications," in *Proceedings of the 9th ACM SIGCOMM Conference on Internet Measurement Conference*, ser. IMC '09. New York, NY, USA: ACM, 2009, pp. 280–293. [Online]. Available: http://doi.acm.org/10.1145/1644893.1644927

[5] F. Qian, Z. Wang, Y. Gao, J. Huang, A. Gerber, Z. Mao, S. Sen, and O. Spatscheck, "Periodic transfers in mobile applications: Network-wide origin, impact, and optimization," in *Proceedings of the 21st International Conference on World Wide Web*, ser. WWW '12. New York, NY, USA: ACM, 2012, pp. 51–60. [Online]. Available: http://doi.acm.org/10.1145/2187836.2187844

[6] Y. Xiao, P. Hui, P. Savolainen, and A. Ylä-Jääski, "Cascap: Cloud-assisted context-aware power management for mobile devices," in *Proceedings of the Second International Workshop on Mobile Cloud Computing and Services*, ser. MCS '11. New York, NY, USA: ACM, 2011, pp. 13–18. [Online]. Available: http://doi.acm.org/10.1145/1999732.1999736

[7] F. Sailhan and V. Issarny, "Energy-aware web caching for mobile terminals," in *Distributed Computing Systems Workshops, 2002. Proceedings. 22nd International Conference on*, 2002, pp. 820–825.

[8] H. Shen, M. Kumar, S. K. Das, and Z. Wang, "Energy-efficient data caching and prefetching for mobile devices based on utility," *Mob. Netw. Appl.*, vol. 10, no. 4, pp. 475–486, Aug. 2005. [Online]. Available: http://dl.acm.org/citation.cfm?id=1160162.1160171

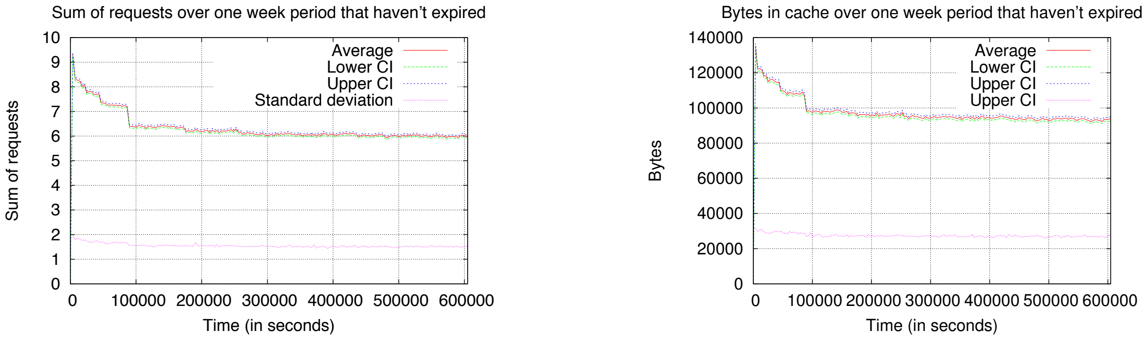
[9] B. Thyamagondlu, V. Chu, and R. Wong, "A bandwidth-conscious caching scheme for mobile devices," in *Big Data (BigData Congress), 2013 IEEE International Congress on*, June 2013, pp. 78–85.

[10] P. Meenan, "How fast is your web site?" *Queue*, vol. 11, no. 2, pp. 60:60–60:70, Mar. 2013. [Online]. Available: http://doi.acm.org/10.1145/2436696.2446236

[11] G. P. Perrucci, F. H. P. Fitzek, and J. Widmer, "Survey on energy consumption entities on the smartphone platform," in *Vehicular Technology Conference (VTC Spring), 2011 IEEE 73rd*, May 2011, pp. 1–6.

[12] S. A. Krashakov, A. B. Teslyuk, and L. N. Shchur, "On the universality of rank distributions of website popularity," *Computer Networks*, vol. 50, no. 11, pp. 1769 – 1780, 2006. [Online]. Available: http://www.sciencedirect.com/science/article/pii/S1389128605002513

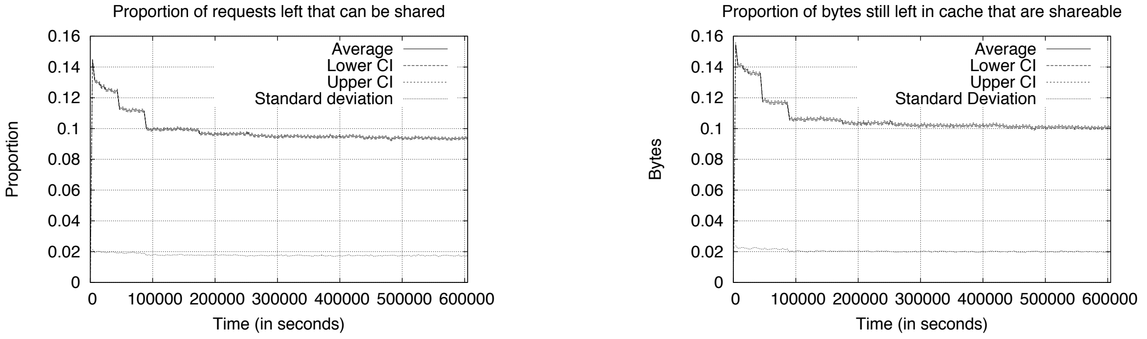
[13] G. Anastasi, M. Conti, E. Gregori, and A. Passarella, "Performance comparison of power-saving strategies for mobile web access," *Performance Evaluation*, vol. 53, no. 3, pp. 273–294, 2003.



(a) Requests

(b) Bytes

Fig. 6. Simulation results for the number of responses and bytes that are in the transfered mobile device cache.



(a) Requests

(b) Bytes

Fig. 7. Simulation results for the attained savings for the number of responses and bytes by transferring to the mobile device cache.