Study of Web Acceleration Technique " Event Logger for Firefox (ELF)"

[1]Indumati, [2]Dr. Pankaj Verma

[1]Research Scholar, [2]A.P., Department of CSA, Niilm University

Abstract : The ever increasing use of World Wide Web in the existing network, results in poor erformance. Several techniques have been developed for reducing web traffic by compressing the size of the file, saving the web pages at the client side, changing the burst nature of traffic into constant rate etc. No single method was adequate enough to access the document instantly through the Internet. In this paper, we will analyze Event Logger for Firefox (ELF)

ISSN : 2348-5612 © URR

Key Words : www, ELF, Firefox etc.

Introduction : The exponential rate of growth of World Wide Web (WWW) has led to an increase in Internet traffic and degradation in user-perceived latency. With the continuing growth of Internet, web users are suffering from the network congestion and server overloading. Content simplification and data compression techniques are used to reduce the amount of bytes sent over the network. Since these techniques cannot be used for much reduction in bandwidth consumption, alternate schemes such as web caching, web cache sharing, web server pushing, browser initiated server pushing are developed. But these techniques have not yet attained the satisfactory results. Web caching and web cache sharing are unsatisfactory techniques due to dynamic documents, CPU overhead, memory spending for caching, administrative overhead etc. In web server pushing technique, it is difficult for a web content provider to know the proper place to push documents.

In the browser initiated server pushing, the server does not know what is in the client cache. Without considering the client’s cache, the web server pushes more images than needed. The user interacts only with the web browsers to fetch the web pages. So the browser can store the history of each user behavior and it can take intelligent decision depending upon the user behavior and traffic condition with the help of intelligent agents. In the proposed adaptive traffic reduction Motivation

Techniques :

The technical challenges for establishing Internet access are relatively clear. Beyond improving network infrastructure, many ideas and several systems have been implemented to address connativity challenges. Web acceleration techniques such as caching, compression, and prefetching have been applied in the past to address exactly these bandwidth and latency constraints. More aggressive techniques such as caching stale pages and client-side prefetching have been proposed previously, but only evaluated in disconnected or intermittently connected networks.

Evaluation of the ELF

In addition to understanding web usage in the context of a peri-urban school in an emerging region, we were interested to explore how much web acceleration techniques recommended by prior work would actually improve the user experience. In addition to logging we implemented several of these techniques in ELF.

Caching without Expiration

The first web acceleration technique that we implemented was to sacrifice freshness for speed and offline availability. The basic idea behind this technique is to allow the presentation of potentially stale cached content to users to reduce the amount of traffic transferred over the network. This idea was recommended 4 years ago and deployed in the C-LINK system in Nicaragua , however, until now it is not been bundled as an easily deployable browser extension. To implement this technique, ELF performs several modifications to Firefox's default behavior.

• ELF sets the check doc f requency to 2 meaning Firefox checks online for an updated version of a file only if it is not in the cache. The Firefox default is 3, which means Firefox checks for an updated file if the cached version is expired.

• ELF changes the cache behavior by extending the cache duration of each file by an extra 30 minutes. The "nocache" headers in all web responses are also ignored and the expiration is set to 30 minutes. While the expiration notices are ignored by ELF (as per the previous point), they are still important for maintaining the correct cache eviction order used by Firefox.

Web Prefetching Effectiveness

Predicting future web accesses and prefetching them ahead of user requests may reduce user response time. While high-speed networks in the developed world have practically eliminated the utility of prefetching, its potential benefits are still critical for developing world clients

with long-latency, low-bandwidth Internet accesses. In this section, we evaluate the effectiveness of prefetching techniques for developing world using the following metrics:
(1) hit rate—the number of hits to the prefetching cache divided by the total number of accesses; (2) overhead rate—the number of downloaded pages due to prefetching divided by the number of accesses;
(3) use rate—the proportion of prefetched pages that are indeed accessed later

Offline Browsing

The third and final technique that we implemented was offline browsing2. While some form of offline browsing has been implemented previously there has not been a formal evaluation of its effectiveness. To support offline browsing, ELF modifies Firefox in the following ways:

• ELF sets the disk cache size to 500 MB uniformly across all machines to increase the amount of offline browsing possible. The default cache size for our version of Firefox was 50 MB. Any existing files in the cache were left in place, and no changes to the other cache size settings were made.

• ELF modifies the “Cache-Control” header for all web responses in Firefox so that the “max-age=31536000”, which indicates that files should be evicted only after they are 1 year old.

• ELF checks the machine is online or offline by sending a HEAD request to a popular search portal every 5 minutes. If the host was unreachable, ELF would put Firefox in offline mode.

• In offline mode, ELF modifies the rendering of all pages using a style sheet to indicate the links that refer to pages that exist in the cache to assist users in avoiding dead links while browsing offline.

Conclusion

From our results, freshness vs speed is a good tradeoff for users behind constrained network connections. We realize that this tradeoff violates RFC2616 , but, as suggested by prior work, users in these settings are unlikely to be interested in minute changes in content on sites other than news, webmail, and Facebook . We also found that prefetching, while beneficial, could be further improved, and client-end prefetching algorithms are an interesting avenue for future work. We found that offline cache browsing is only useful for visiting previously viewed pages. While some prior work has suggested that less experienced web users perform history-based browsing , this appears to be a corner case in our setting. We hypothesize that another shortcoming of offline browsing is that the rate pages are prefetched or cached cannot keep up with the pages browsed while offline. Under a constrained setting, a user will eventually get a cache miss and fail to make progress.

Finally, there were two optimizations that could have been useful that we did not implement, but have been studied independently. Collaborative caching across networks behind upstream bottlenecks such as ours could have been a way to gain the greater benefits of caching without access to the gateway proxy . Caching partially downloaded or chunks of content could also be useful particularly for YouTube videos and highly synergistic with cooperative caching. We hope to add these caching optimizations to our extension in the next version. While the prevalence of video may appear to undermine the benefits of our optimizations, we note that the video downloads are too slow to be viewed in real-time. This suggests that people are loading the videos first and watching them after completion, and in terms of user experience, these load times are not as disruptive as those during a more real-time task such as web browsing.

References :

1. D. L. Johnson, V. Pejovic, E. M. Belding, and G. van Stam. Traffic characterization and internet usage in rural Africa. In 20th Int’l Conf. on World Wide Web (WWW), 2011.
2. V. N. Padmanabhan and J. C. Mogul. Using predictive prefetching to improve World Wide Web latency. ACM SIGCOMM Computer Communication Review, 1996.
3. R. Patra, S. Nedevschi, S. Surana, A. Sheth, L. Subramanian, and E. Brewer. WiLDNet: Design and implementation of high performance WiFi based long distance networks. In 4th USENIX Symp. on Networked Systems Design and Implementation (NSDI), 2007.Tridgell. Efficient Algorithms for Sorting and Synchronization. D thesis, The Australian National University, 1999.
4. T. Wood, K. Ramakrishnan, P. Shenoy, and J. Van der Merwe. CloudNet: dynamic pooling of cloud resources by live WAN migration of virtual machines. In 7th ACM Conf. on Virtual Execution Environments (VEE), 2011.
5. World summit on the information society (Geneva 2003—Tunis 2005). What’s the state of ICT access around the world?
6. http://www.itu.int/wsis/tunis/newsroom/stats/. F. Zhang, J. Chen, H. Chen, and B. Zang. CloudVisor: Retrotting protection of virtual machines in multi-tenant cloud with nested virtualization. In 23th ACM Symp. on Operating Systems Principles (SOSP), 2011.