

Hidden Challenges on Web Application Design

Eric Freeman Katia Passos Cerise Wuthrich Catherine Stringfellow
Midwestern State University
Wichita Falls, TX 76308

Abstract

An existing mechanism that improves the quality of service perceived by Web users is web caching, in which intermediary systems are used to temporarily store information that may be retrieved by multiple users multiple times. Web caching produces a variety of results that software engineers do not consider. However, ignoring the existence of web caches at the application design stage may introduce critical errors in the behavior of a system. In this paper, a discussion on how web caching may affect the application is presented, as well as basic ideas on how to design an application considering those hidden resources.

Introduction

A potential problem on web caching is the currency of the information of a stored Web page [1,4]. An analysis of a set of interactive Web applications will show that designers prefer using scripts to assemble Web pages after dynamically evaluating a user's request. This approach results in dynamic Web pages that are not usually cached, and consequently do not benefit from the performance improvement provided by the cache [2].

Many studies have been conducted in this area to decide when to cache a dynamic page [5]. The hypertext transfer protocol (HTTP) defines cache management parameters that can be used to specify if a page is 'cacheable' or not. However, this option eliminated the cache benefits. New extensions to software engineering methods allow the designer to anticipate the use of communication networks and multiple systems [3, 4].

Web application challenges

Web caching is usually transparent to the user and to the application designer, except for the possible improvement in response time. The application designer, when planning the development of a system, usually will not have enough information to judge if a web cache is involved. To verify the possibility of caching errors, an online wireless testing application was developed to allow students to take multiple-choice tests (MCT) through a PDA device.

The application was tested using simulators of existing PDA browsers, which initially assumed that any page is cacheable. The MCT was taken twice without disconnecting and supplied with exactly the same answers. Before retaking the test, the server database was

updated and the correct answer for one question was changed. Actual results showed the same score, because cached pages were used. The same experiment was conducted with simulators that fetched pages from the local cache only upon using the *Back* button. In this case the tests worked correctly, since the simulator had not cached any pages with dynamic content.

To improve the correctness of the first simulation, meta tags were added to the HTTP header of the page code, specifying a zero value for maximum age. This action was equivalent to declare the page non-cacheable. In this case all pages were retrieved from the server. However, the user's response time deteriorated.

Conclusion

This study helped to confirm the uncertain behavior of the caching systems. Research in caching dynamic pages suggests that future changes will be implemented in systems for which the developer will not be able to prepare. The use of unique parameters in page requests seems to be the more stable and reliable solution at this point and worked well in experimental trials with a slight impact in performance.

Acknowledgements

This work was partially supported by the Texas Advanced Research Program under Grant No. 003656-0108b-2001.

References

- [1] M. Arlitt, et al, "Evaluating Content Management Techniques for Web Proxy Caches," *ACM SIGMETRICS*, 27(4), 2000, 3-11.
- [2] G. Barish and K. Obraczka, "World Wide Web Caching: Trends and Techniques," *IEEE Communications*, 38(5), 2000, 178-184.
- [3] A. Bouguettaya, et al., "Supporting Dynamic Interactions among Web-Based Information Sources," *IEEE Trans. on KDE*, 12(5), 2000, 779-801.
- [4] R. J. A. Buhr, "Use Case Maps as Architectural Entities for Complex Systems," *IEEE Transaction on Software Engineering*, 24(12), 1998, 1131-1155.
- [5] K. S. Candan, et al., "Enabling Dynamic Content Caching for Database-Driven Web Sites," *ACM SIGMOD 2001*, 30(2), May, 2001, 532 - 543.
- [6] L. Rizzo and L Vicasano, "Replacement Policies for a Proxy Cache" *IEEE/ACM Trans. on Networking*, 8(2), 2000, 158-170.