Efficient Web Change Monitoring with Page Digest

David Buttler
Center for Applied Scientific Computing
Lawrence Livermore National Laboratory
Livermore, CA 94550
buttler1@llnl.gov

Daniel Rocco
College of Computing
Georgia Institute of Technology
Atlanta, GA 30332
rockdj@cc.gatech.edu

Ling Liu
College of Computing
Georgia Institute of Technology
Atlanta, GA 30332
lingliu@cc.gatech.edu

ABSTRACT

The Internet and the World Wide Web have enabled a publishing explosion of useful online information, which has produced the unfortunate side effect of information overload: it is increasingly difficult for individuals to keep abreast of fresh information. In this paper we describe an approach for building a system for efficiently monitoring changes to Web documents. This paper has three main contributions. First, we present a coherent framework that captures different characteristics of Web documents. The system uses the Page Digest encoding to provide a comprehensive monitoring system for content, structure, and other interesting properties of Web documents. Second, the Page Digest encoding enables improved performance for individual page monitors through mechanisms such as short-circuit evaluation, linear time algorithms for document and structure similarity, and data size reduction. Finally, we develop a collection of sentinel grouping techniques based on the Page Digest encoding to reduce redundant processing in large-scale monitoring systems by grouping similar monitoring requests together. We examine how effective these techniques are over a wide range of parameters and have seen an order of magnitude speed up over existing Web-based information monitoring systems.

Categories and Subject Descriptors

H.3.4 [**Systems and Software**]: Alert services

General Terms

Algorithms, Performance

Keywords

Scalability, Web document monitoring, Document storage

1. INTRODUCTION

The World Wide Web offers a unique publishing medium that enables information broadcast with few of the traditional barriers to widespread communication. This freedom has fostered growth of online communities that publish information on a near limitless array of topics. Automatic Web change monitoring provides several compelling advantages even for simple scenarios. First, automatic systems remove the burden of monitoring from the user, allowing them to concentrate on other efforts while being assured of receiving timely notification when an interesting change occurs. Second, Web change monitors can track many different sources simultaneously: users can handle more data effectively, making them more productive and increasing the quality of their decisions.

We present an automatic Web change detection system that provides a mechanism for monitoring Web information sources, offering semantically rich data processing services and a fine granularity change detection with expressive power. This paper describes the system architecture, addressing areas that offer opportunities for optimization. The design provides a framework for flexible and scalable Web change monitoring through the use of: *efficient data management* using the Page Digest format [2]; *rich processing constructs* for targeted content and structure; and *grouping* popular pages, eliminating redundant computation.

2. ARCHITECTURE

The general architecture of our Page Digest sentinel system is a third-party change monitoring system handling queries and data on behalf of users, while maintaining independence from the data sources. The core component is the sentinel manager, which installs and removes sentinels, triggers sentinel evaluation at the user specified interval, and intelligently batches queries to maximize processing efficiency over popular targets.

The sentinel manager's central role in our Web change monitoring system mandates efficient processing of sentinels to ensure system scalability. One of the most important optimizations in the sentinel manager is the grouping of related sentinels together to minimize redundant processing and I/O.

Page Digest Sentinels. An important problem affecting the scalability of any system that interacts with the Web is the processing of standard HTML and XML Web documents. Present Web document formats are redundant and intermix the structure, tag names, attributes, and content. These drawbacks lead us to consider an alternate data encoding that would facilitate fast algorithms and large-scale optimizations. The Page Digest Web document encoding [2] increases processing efficiency in our Web document monitoring system by providing direct access to the different semantically interesting characteristics of a Web document while eliminating tag redundancy.

2.1 Web Document Monitoring

We now consider the challenges of a Web document monitoring system. The first issue is that of user interaction, which can be divided into two subproblems: query specification and user notification. The second issue is that of data management, which includes document storage, change evaluation, versioning, and effective use of compute resources. Here we only examine the query specification and data management problems.

Update Semantics. There are two elemental types of changes to, and sentinels over, Web documents: content changes and structural changes. Content changes include any change to text and changes to hyperlinks or images. In contrast, structural changes modify the tag structure of the document and alter the relationships between document elements.

WWW2004, May 17–22, 2004, New York, New York, USA.
ACM 1-58113-912-8/04/0005.

We expand the semantic flexibility of these two basic change types along several refinement axes. First, structural changes can be restricted to a particular logical group of structural elements, such as attribute alterations. Second, users typically desire to “hone in” on a small part of a document, leading to refinement by location. Third, an update may be triggered only if it satisfies some regular expression pattern or structure expression. Fourth, a change may be defined in terms of combinations of the two basic change types with any of the refinement modifications present. Finally, a change may be detected over a custom interval, although our implementation restricts this to be a multiple of a minimum polling interval.

2.2 Page Digest Sentinel Processing

The challenge in optimizing a monitoring system is in determining the primary costs and implementing effective schemes to minimize those costs. Those costs can be categorized as network costs, data management costs, and processing costs. Network costs are beyond the control of a third-party system. Data management costs are partially addressed by the Page Digest format; in addition, our system maintains document signatures and sentinels in memory, eliminating a large fraction of local I/O costs. Here we focus on local processing costs incurred during the evaluation of sentinels and describe how we leverage the Page Digest data structure to alleviate the costs.

The implementation of sentinels in our Web change monitoring system is as a sentinel manager, which is responsible for processing sentinel queries. Since the goal of our system is scalability, the implementation seeks to maximize throughput and may sacrifice individual sentinel latency if necessary. We maximize throughput in two main ways: minimizing redundant network access and minimizing local processing.

Shortcut Techniques. *Preprocessing* Depending on the type of sentinel and the cost, documents may have signatures created to shortcut processing when no change occurs. *Location masking* Sentinels may mask out the interesting portion of a document, reducing the computation required for all change detection algorithms.

Content Sentinels. Text change detection operates over the document’s content list. The algorithms match user-defined regular expressions or keyword phrases directly over the content container, automatically bypassing other parts of the document. The separation of document components in the Page Digest encoding allows text change detection to operate efficiently over the text without introducing extraneous computational overhead for parsing different document elements during the search.

Structure Sentinels. The Page Digest encoding allows structure sentinels to avoid loading the entire document into memory for processing. In many cases, the structure of an HTML document may be small enough to maintain in memory with the sentinel. Structure information can be used to dramatically speed up the execution of generic tree-to-tree change detection algorithms in the case where there are few or no differences between the document’s structure. In addition, the structure information may be used to focus in on specific features, such as links, for fast scanning.

2.3 Multiple Sentinel Processing

By combining related requests, we are able to minimize network costs and reduce processing costs with only a modest overhead for group maintenance. Sentinel grouping allows a single fetch for all sentinels over a document. This optimization assumes that all these sentinels expect updates at the same interval. For groups where this is not the case, the sentinel grouping process temporarily removes sentinels that are not due for notification from the group.

Coupled with the Page Digest encoding, grouping allows the system to perform change detection only once for each group, efficiently reusing the change information for individual sentinels. Grouping also eliminates redundant processing by executing more general sentinels in the group before more specific ones.

Group processing begins when the sentinel manager is triggered to check for changes that affect the group. The sentinel manager retrieves all sentinels installed on the current URL and selects the active set for this firing interval. The new version is retrieved from its source, hashed to a signature, and if changed the previous version is loaded from disk.

The sentinel manager computes an annotation array, marking what has changed, then examines the locations in the annotation array specified in each sentinel’s location mask. Comparison begins with the sentinels in the minimal covering subset, which is the smallest subgroup of sentinels that covers every location of interest to the group for each particular change type. If none of the sentinels in this subgroup detects a change, no further processing is needed.

2.4 Experimental Summary

We performed several experiments to measure the system against other systems and techniques, and to demonstrate the power of our optimizations. Our first experiment compared the overall performance of the sentinel processing system to the WebCQ system [1]. Each of the novel characteristics in the Page Digest sentinel system addresses performance bottlenecks from this system, allowing us to achieve between one and two orders of magnitude speedup in processing time. Our second experiment examined sentinels that were grouped in a Zipf-like distribution over common Web pages. By combining processing between similar monitoring requests we were able to improve the execution time by 50-70%. The third experiment examined the cost of grouping large numbers of sentinels, showing that the cost is amortized over the size of the group, costing less than loading a document into memory, and approximately the cost of computing an MD5 signature for the document. The final experiment demonstrated the performance increase of using the Page Digest format over using the more popular DOM tree, at less than half the cost for the same sentinel type.

3. CONCLUSION

We have presented a new mechanism for detecting changes to HTML and XML documents based on the Page Digest encoding, focusing on providing standard change detection mechanisms, operators for more advanced change detection operators, techniques to monitor different aspects of a page, including content, tag types, attributes, and structure, and, finally, a set of unique optimization techniques based on the Page Digest encoding that dramatically improves the performance of the system.

Acknowledgments: This work was partially performed under the auspices of the U.S. Department of Energy by University of California Lawrence Livermore National Laboratory under contract No. W-7405-ENG-48. UCRL-CONF-202469.

4. REFERENCES

[1] L. Liu, C. Pu, and W. Tang. WebCQ: Detecting and Delivering Information Changes on the Web. *Proceedings of the International Conference on Information and Knowledge Management*, November 2000.

[2] D. Rocco, D. Buttler, and L. Liu. Page digest for large-scale web services. In *Proceedings of the IEEE Conference on Electronic Commerce*, 2003.