

Scalable and Secure Data Collection: Fault Tolerance Considerations*

W. C. Cheng[†] L. Cheung[†] C.-F. Chou[‡] L. Golubchik[§] Y. Yang[†]

Abstract

Data collection, or *uploading*, is an inherent part of numerous digital government applications. In this poster we present our recent research directions in the development of Bistro, a scalable and secure architecture designed for collection of data over the Internet for digital government applications.

1 Introduction

Hotspots are a major obstacle to achieving scalability in the Internet; they are usually caused by either high demand for some data or high demand for a certain service. At the application layer, hotspot problems have traditionally been dealt with using some combination of increasing capacity, spreading the load over time and/or space, and changing the workload.

These classes of solutions have been studied in the context of applications using the following types of communication: (a) one-to-many (data travels primarily from a server to multiple clients, e.g., web download, software distribution, video-on-demand); (b) many-to-many (data travels between multiple clients, through either a centralized or a distributed server, e.g., chat rooms, video conferencing); and (c) one-to-one (data travels between two clients, e.g., e-mail, e-talk). However, to the best of our knowledge ours is the first work on making applications using *many-to-one* communication scalable and efficient; existing solutions, such as web based uploads, simply use many independent one-to-one transfers. This corresponds to an important class of applications, whose examples include a large number digital government applications.

Specifically, government at all levels is a major *collector* and provider of data, and there are clear benefits to disseminating and collecting data over the Internet, given its existing large-scale infrastructure and wide-spread reach in commercial, private, and government domains. In this project, we focus on the *collection of data over the Internet*, and specifically, on the *scalability* issues which arise in the context of Internet-based massive data collection applications. By data collection, we mean applications such as Internal Revenue Service (IRS) applications with respect to electronic submission of income tax forms. Briefly other such applications are as follows. The Integrated Justice Information Technology Initiative facilitates information sharing among state, local, and tribal justice components. An integrated (global) information sharing system involves collection, analysis, and dissemination of criminal data. Clearly, in order to facilitate such a system one must provide a *scalable* infrastructure for collection of data. Furthermore, a number of government agencies (e.g., NSF, NIH) support research activities, where the funds are awarded through a grant proposal process, with deadlines imposed on submission dates. The entire process involves not only submission of proposals, which can involve fairly large data sizes, but also a review process, a reporting process (after the grant is awarded), and possibly a results dissemination process. All these processes involve a data collection step. Lastly, digital democracy applications, such as online voting during federal, state, or local elections, constitute another set of massive upload applications. Of course, there are numerous other examples of digital government applications with large-scale data collection needs.

*This work is supported by the NSF Digital Government Grant 0091474. More information about the Bistro project can be found at <http://bourbon.usc.edu/iml/bistro>.

[†]Computer Science Department, University of Southern California, Los Angeles, CA. Email: bill.cheng@acm.org, {lccheung, yangyan}@usc.edu.

[‡]Department of Computer Science and Information Engineering, National Taiwan University. Email: ccf@csie.ntu.edu.tw.

[§]Computer Science and EE-Systems Department, IMSC, and ISI, University of Southern California, Los Angeles, CA. Email: leana@cs.usc.edu.

Our past work proposed Bistro [1, 3], a framework for building scalable and secure wide-area digital government *upload* applications. A summary of main advantages of the Bistro architecture is as follows: (1) hotspots can be eliminated around the server because the transfer of data is decoupled from making of the deadline, (2) clients can receive good performance since they can be dispersed among many bistros and each one can be direct to the “best” bistro for that client, and (3) the destination server can minimize the amount of time it takes to collect all the data since now it is in control of when and how to do it (i.e., Bistro employs a server pull).

Our past work focused on performance and security issues. This poster focuses on recent research directions on fault tolerance issues related to large-scale data collection within the Bistro framework.

2 Fault Tolerance Directions

The security mechanisms in the Bistro upload protocols [2] guarantee integrity and privacy of the data being uploaded. However, to improve the performance characteristics of our architecture, it is still desirable to provide mechanisms and policies for ensuring that data will not have to be retransmitted due to losses or temporary unavailable which could occur due to failures or malicious behavior of various system components.

To this end, our current work focuses on augmenting the original Bistro architecture with appropriate fault tolerance and redundancy mechanisms and policies, where the amount of redundancy and degree of fault tolerance depends on the application and the reliability characteristics of the system components. In particular, we are interested in using a combination of erasure codes and checksums to enhance the performance and reliability of the Bistro system.

Several interesting questions arise in this context, including (a) which fault tolerance schemes should be used in Bistro, (b) how much redundancy in communication bandwidth and temporary storage is needed, (c) what is the effect on the performance of the upload system when fault tolerance protocols are introduced, and (d) what is the effect of providing fault tolerance on each step of the Bistro protocol.

Our goal in this work is to maintain comparable performance to that of a system without fault tolerance mechanisms and to reduce the overhead attributed to fault tolerance mechanisms (such as storage and network bandwidth overheads) as much as possible. We are currently in the process of completing the fault tolerance protocol design as well as the corresponding analysis and simulation studies which will allow us to address the above raised issues.

3 Conclusions

Our work thus far indicates that efficient, scalable, secure, and fault-tolerant data collection is possible over the public Internet for digital government applications. We believe that much work remains to be done in bringing these ideas to their full potential as well as in investigating their applicability to a broader class of digital government applications.

References

- [1] S. Bhattacharjee, W. C. Cheng, C.-F. Chou, L. Golubchik, and S. Khuller. Bistro: a platform for building scalable wide-area upload applications. *ACM SIGMETRICS Performance Evaluation Review (also presented at the Workshop on Performance and Architecture of Web Servers (PAWS) in June 2000)*, 28(2):29–35, September 2000.
- [2] W. C. Cheng, C.-F. Chou, L. Golubchik, and S. Khuller. A secure and scalable wide-area upload service. In *Proceedings of the 2nd International Conference on Internet Computing, Volume 2*, pages 733–739, June 2001.
- [3] W.C. Cheng, C.F. Chou, L. Golubchik, S. Khuller, and H. Samet. Scalable data collection for internet-based digital government applications. In *1st National Conference on Digital Government Research*, pages 108–113, Los Angeles, CA, May 2001.