WebFS: A Global File System for Fine-Grained Sharing

Amin Vahdat, Michael Dahlin, Paul Eastham, and Thomas Anderson
http://now.cs.berkeley.edu/WebOS

The growth of the Internet and World Wide Web has made wide area, large scale file sharing the common case. Unfortunately, the existing HTTP standard suffers from a number of inefficiencies which are currently beginning to strain the underlying Internet infrastructure. Further, missing HTTP functionality prevents the development of a new class of collaborative applications. To address these limitations we are building WebFS, a global filesystem designed to support large-scale read/write sharing.

While existing Web protocols have allowed for a quantum leap in wide-area collaboration, they do not fully address some key functionality, including: i) an efficient cache-coherent global name space, ii) authentication for read/write access, iii) multiple application-specific caching policies, and iv) a well-defined failure and recovery model. We aim to address these limitations by building WebFS, a distributed cache-coherent global filesystem. We believe WebFS can greatly simplify the design and implementation of a number of existing collaborative applications. We are currently using WebFS to prototype a shared whiteboard, a scalable chat service [Yoshikawa et al. 1997], Rent-A-Server [Vahdat et al. 1996], and a remote file backup tool.

The initial WebFS design provides the following functionality to support wide area collaborative applications:

- **Multicast for Cache Consistency**: WebFS associates a multicast address [Deering 1991] with each open file on a site by site basis. Thus, applications are delivered invalidations and updates more efficiently than HTTP unicasts. Another benefit is that a file's host need not maintain a directory of sites caching a file.
- **Tolerance to Network and Host Failure**: While systems such as Coda [Kistler & Satyanarayanan 1992] and Bayou [Terry et al. 1995] address disconnected and weakly connected operation in the local area, we believe that wide area collaboration will provide a number of unique challenges. Multiple distributed writers will often constitute the common case and the network performance among distributed users sharing a service can vary significantly. One approach we are considering borrows from Scalable Reliable Multicast [Floyd et al. 1995]. In this scheme, a globally unique sequence number is associated with each file update, allowing sites to detect lost updates and multicast requests for missing data.
- **Application-Specific Caching Policies**: Different distributed applications require varying degrees of cache consistency guarantees. We plan to implement a number of caching schemes to explore the tradeoffs associated with client/server performance and consistency guarantees.
- **Authentication**: Webfs combines Public key cryptography [Rivest et al. 1978] with unique nonce exchange to authenticate users to remote sites. WebFS provides access control lists for read, write, and execute access at the granularity of individual users.

References

[Deering 1991] S. Deering. "Multicast Routing in a Datagram Internetwork". PhD thesis, Stanford University, December 1991.

[Floyd et al. 1995] S. Floyd, V. Jacobson, S. McCanne, C.-G. Liu, and L. Zhang. "A Reliable Multicast Framework for Light-weight Sessions and Application Level Framing". In *IEEE/ACM Transacions on Networking*, November 1995.

[Kistler & Satyanarayanan 1992] J. J. Kistler and M. Satyanarayanan. "Disconnected Operation in the Coda File System". *ACM Transactions on Computer Systems*, 10(1):3–25, February 1992.

[Rivest et al. 1978] R. L. Rivest, A. Shamir, and L. Adelman. "A Method for Obtaining Digital Signatures and Public-Key Cryptosystems". In *Communications of the ACM*, volume 21, February 1978.

[Terry et al. 1995] D. B. Terry, M. M. Theimer, K. Petersen, A. J. Demers, M. J. Spreitzer, and C. H. Hauser. "Managing Update Conflicts in Bayou, a Weakly Connected Replicated Storage System". In *Proceedings of the Fifteenth ACM Symposium on Operating Systems Principles*, pp. 172–183, 1995.

[Vahdat et al. 1996] A. Vahdat, M. Dahlin, and T. Anderson. "Turning the Web into a Computer". May 1996. See http://now.cs.berkeley.edu/WebOS.

[Yoshikawa et al. 1997] C. Yoshikawa, B. Chun, P. Eastham, A. Vahdat, T. Anderson, and D. Culler. "Using Smart Clients to Build Scalable Services". In *Proceedings of the USENIX Technical Conference*, January 1997.