

# Moderated collaboration to modify shared files among wireless users

Surendar Chandra (surendar@nd.edu) and Nathan Regola (nregola@nd.edu - student author)  
*University of Notre Dame*

The goal of this research is to develop practical collaboration mechanisms that allows any group members to modify shared files. Our system is designed to support far more conflicting updates than was assumed by systems such as Coda [3] and Ficus [2]. Resource rich laptops were observed to become the primary computing platform of contemporary users. Hence, we design our system for wireless users. The ubiquity of high speed wireless networks allow these users to operate from a variety of locations: home, work and from a *third place*. In each of these locations, users encounter different kinds of local participants while still communicating with their global partners who also use wireless networks. Our empirical analysis of the wireless availability behavior of a large number of wireless users in a variety of communities: academia, corporate lab, conference attendees and in a city-wide hotspot showed a trend towards smaller user session lengths as well as longer durations between sessions with observable node churn. An investigation of the significance of these observations show that the user availability patterns has an detrimental impact on the behavior of prior group collaboration systems. As compared to desktop users, wireless users appear to stay online for just enough time to complete their own tasks without fruitfully participating in providing better sharing for other group members. We observed that a mandatory locking scheme rejects many updates because of simultaneous requests. Similarly, AFS like *last writer wins* semantics causes a large number of inconsistent updates with an observable loss of update causality. Also, asynchronous propagation systems such as Bayou [1] experience a large number of update conflicts and transaction roll-backs. More importantly, the worst performance in all these systems was observed during the peak availability durations; these systems fail when all the group members were available.

We showed the inadequacy of prior systems that maintain a single consistent copy of the shared documents. Instead, **we develop a moderated collaboration mechanism that maintains multiple copies of the shared object**. We maintain one updateable copy of the shared content on each group member's node. We also hoard read-only copies of each of these updateable copies in any interested group member's node. For a group of size  $n$ , we can potentially maintain one updateable copy (the author) and  $(n - 1)$  read-only copies (from

other users in the collaboration group). Given the vast improvements in laptop storage (a 500GB 2.5" hard disk retails for less than USD\$120), extra copies are a reasonable overhead. Group members can reduce the replica maintenance overhead by explicitly specifying the group members whose contents are replicated. Each author manually moderates and incorporates modifications from other group members using these read-only replicas. Moderation operations are similar to manual reconciliation operations in Coda and Ficus except that the moderator incorporates updates from all the other group members into their copy. The various versions of the shared document will eventually converge through independent moderation operations. The system automatically logs the provenance of all causal reads of contents from other replicas into the author versions. These provenance records allows any users to independently decide whether their updates had been incorporated into the final version of a particular file.

Using the wireless access traces, we show that our system avoids many of the problems of prior systems. Except for small groups, our distributed approach achieves similar performance as a server based system. We have built a prototype of our system using Git, an open source distributed version control system and the Fuse userspace file system. Git is designed to be fast and efficient (space and time) and we inherit these advantages. Preliminary benchmarks show that our system achieves performance similar to a fuse file system. This system is in regular use within our group. We are in the process of releasing it to a larger audience.

## References

- [1] A. Demers, K. Petersen, M. J. Spreitzer, D. Terry, M. Theimer, and B. Welch. The bayou architecture: support for data sharing among mobile users. In *Workshop on Mobile Computing Systems and Applications*, pages 2–7, Santa Cruz, CA, Dec. 1994.
- [2] T. W. Page, R. G. Guy, J. S. Heidemann, D. Ratner, P. Reiher, A. Goel, G. H. Kuenning, and G. J. Popek. Perspectives on optimistically replicated peer-to-peer filing. *Software—Practice and Experience*, 28(2):155–180, February 1998.
- [3] M. Satyanarayanan, J. J. Kistler, P. Kumar, M. E. Okasaki, E. H. Siegel, and D. C. Steere. Coda: A highly available file system for a distributed workstation environment. *IEEE Transactions on Computers*, 39(4), Apr. 1990.