FILL IN

Paul Elliott
University of Oregon
paule@cs.uoregon.edu

David Stevens
University of Oregon
dstevens@cs.uoregon.edu

Abstract

FILL IN

1 Project Idea

1.1 Scope/Topic

We look into [file hosting services (Dropbox) — revision systems (Github)], which is a popular system for [collaborative work, remote file storage, FILL IN].

1.2 Problem

These services/systems work well with centralized storage, but keeping all data at a central server or group of servers:

1. Increases risk of privacy issues
2. Central point of failure

[FILL IN more from SOUP and elsewhere]

1.3 Solution

Distribute data storage for these [file hosting services — revision systems] across all users, by storing replicas on individual user machines.

-Punt on security: SHA and PKI and everything else those schizoid nerds have come up with

1.4 Questions

(Given that nodes do not know global state—-and are dealing with a dynamic distributed system–nodes may come and go, nodes may crash, multiple users may work on the same data)

1. Timing/Synchronicity:
   -How do we handle distributed updates of replicas, or rather how do we synchronize updates?
   -How do we synchronize updates
2. Availability vs Network Overhead:
   -How do we handle node failures? How do we update data when a node returns?
   -How do we ensure that a node gets updated/current data?
3. Group membership:
   -How do we keep track of which nodes have replicas?

2 Related Work

(synchronicity/replication strategies–conservative or optimistic–in DS, issues with DropBox, open source revision—file hosting) (FILL IN)

3 Design

Readers and Writers Messages: -Write (create, push update) -Read (pull update) -Join (group, project—revision ?) Version numbers? Synch strategy: -conservative or optimistic?

4 Evaluation

Compare to [DropBox — Github — other open source system]

1. Testing Platform: lossy network
2. Parameters:
   - Types of users:
     -lenient (just want a more current copy of the data)
     -strict (want most up-to-date copy of the data)
   - Environment:
     -unstable (significant node dynamics)
     -stable
3. Metrics:
   - Available:
     -network overhead?
     -failed pull attempts?
   - Reliable:
     -number of successful up-to-date pulls?
4. Scalability:

5 Timeline

1. This weekend: complete literature review (synchronicity/replication strategies in DS, issues with DropBox, open source revision—file hosting)

   -Each of us: research 1 of each

   -Fill in 1 paragraph summary of any literature, add any ideas to notes

2. Week 8: Design
3. Week 9 (Thanksgiving): Testing/Evaluation
4. Week 10: Presentation, Write paper
5. Final Exam Week: Write paper, turn in paper, have multiple celebratory beers

References