COSC480 Project
Aims and Objectives

Calum O'Hare

1 Aims

The aim of this project is to develop a file synchronisation tool. Similar to Dropbox (and others) its main function should be to keep data synchronised between multiple devices. What makes it different however is it should:

- Be decentralised. It will not necessarily need to be run in "the cloud", there should be no centralised server, just many cooperating client nodes.
- Allow file synchronisation between multiple clients—not just point-to-point between two clients. Clients may be running different operating systems. Clients may run on different networks, with different costs of access (including being disconnected from the Internet at times).
- Allow for fine-grained user control for the majority of the program's functions, *e.g.*, how often, and what, to replicate within different sets of files. 'What' could be filename, filetype, filesize *etc.*
- Show statistics about which files are being replicated, efficiency, cost (bandwidth, disk space). These statistics could also possible lead to a heuristic for when to sync a given file.

Example use case

I like to keep all of the data on my laptop backed up to an external hard drive. The data on my computer that I wish to back up falls in to three main categories: documents, music, and movies. Documents are mostly scripts and programs that I am writing for University or work projects. Documents also include reports for assessment. These documents change very frequently and are very important to me. Often these are small files (but not always). My music collection changes relatively infrequently, files are around ≈5MB and I like to have a relatively current backup of this collection. My movie collection contains fairly large files but I don't need it to be backed up very often as it doesn't change very much and I don't care if I loose a couple of DVDs. Files that I work on at University would be very useful to have on my laptop at home. Files I work on at work mostly stay at work but occasionally I might want to bring something home to work on. The other device I always have with me and may be on one of any given (Wi-Fi or 3G) network at a certain time is my smartphone. I would like to have photos taken on this backed up to either (or both) my laptop and external hard drive.

Some of the files I move around are of sensitive or personal nature and I would prefer not to store them with a third party vendor. I also have different synchronisation requirements for different types of data. An effective file synchronisation tool would be of great use to me personally.

After some preliminary analysis of the available file synchronisation tools I have found a tool called Unison to be a promising starting base for this project. Unison is an open source file synchronisation tool, it supports efficient (*i.e.,* it attempts to only send changes between file versions) file synchronisation between two directories (including sub folders) between two machines (or the same machine).

2 Objectives

Primary

- Preliminary analysis
  - Research into which tools would be best to use to build the system. What is the closest thing out there to what I'm trying to build and how does it work *etc.*
- Program that keeps two directories in sync on the computer(s)
  - First simple step in building the program it just needs to replicate files between two directories.

- Sync multiple directories across multiple nodes
  - The program should be able to sync files between many nodes of a graph where the nodes are devices with files and the edges of the graph are network links that may or may not be up at any given time. The program may need to work out an efficient strategy for propagating changes throughout the entire graph.
- Implement fine-grained control (settings)
  - Provide a way for the user to control what is being replicated and when. Also give them useful information on these choice such as the cost of doing the replication.
- Implement statistics about replication
  - Create a user front end that shows interesting statistics about what replication is happening and when. In a clear and easy to understand way.

Extensions

- Merge files (text)
  - How to deal with files that have been edited in more than one place before the sync occurs. Text files may be able to be berged
- Compatibility across multiple platforms, *e.g.*, MacOS X, Linux, Windows, Android, and iOS.
  - Working on desktop systems is a priority, mobile is a secondary concern.