Hyperwebs: A Peer-to-Peer Topology for Distributed Genealogical Databases

Ronald Crowther and Scott N. Woodfield
Department of Computer Science,
Brigham Young University
Provo, UT 84602 USA

To many, genealogical research is the finding and recording of facts about our family history. However, genealogical research is more effective when when we share our information with others. The advent of the personal computer has allowed us to be efficient recorders of information, but it was the arrival of the internet that has allowed us to more effectively share information.

The client/server model is the predominant model of the internet. Users or clients log onto the internet and access information stored on servers. Some of the most successful and frequently visited servers on the internet are genealogy-oriented. Because of these sites we are able to casually gather and share information in a manner we couldn't even dream of just 10 years ago.

We can do even better. While servers are necessary, they are not the solution to all problems. They are expensive. Users must either pay fees or some benefactor must continually donate substantial amounts of money to support the server. Servers are single point sources of failure. If either by accident or sabotage a server fails, all users are inconvenienced for hours or days. The worst types of failure occur when the owner goes out of business. Then the server, with its corresponding database of information, can disappear forever. Servers are also centers of control. Like a monarch, the owner of the server has complete control over the server. If the owner wishes to charge fees, display ads, or sell personal information, they may do so. The user has little or no control over the future content or policies. Servers also do not scale well. If we double the number of clients we either double the cost of the server or suffer significant degradations in speed and reliability. Servers are good for archival purposes but are not as suitable for timely information sharing. If I discover a new piece of information, I must

upload it to a server and hope that an interested party will login and find that information. The other party typically does not know when information arrives so they must frequently poll the server to see if anything interesting is available. This process of manually uploading and downloading information is frustrating and ineffective.

Peer-to-peer technologies allow us to overcome some of these problems. There is no server, so, except for the software, a peer-to-peer system is "free" compared to a server-based system. There are no single points of failure in a peer-to-peer system. Thus, they are far more reliable and are more resistant to the kind of attacks made against servers. There is no owner to go out of business so peer-to-peer systems can live for long periods of time without external funding. Such systems only die when there is no one interested in the information that was shared among peers. While there must be some form of control in a distributed system, it is usually more democratic than dictatorial. Peer-to-peer systems also scale well. Without the infusion of money, the apparent speed and reliability of servers decreases linearly with the increase in the number of clients. Peer-to-peer systems can increase in reliability, and speed only decreases logarithmically. Peer-to-peer systems also enable the automated sharing of information. New information recorded in one's local database can be automatically disseminated to all interested parties in a matter of seconds.

Peer-to-peer systems do have their weaknesses. They can be more complex than client/server systems. This is especially true from the client or peer's perspective. The technology is new and is not mature. The peer-to-peer systems useful in genealogical research are most beneficial when the peer is continually connected to the internet. This will probably become commonplace in the future but it is not common now, especially for those mature individuals who do the bulk of the genealogy work.

Most of the peer-to-peer software available today is not suitable for our purposes. A major assumption seems to be that peers wish to remain anonymous. Thus the software supports anonymous connections of short duration. Genealogical information is more effectively shared when connections are continuous, fast, and between friends, especially friends we can visit repeatedly. Another problem with

current peer-to-peer systems is lack of scalability. Current systems are based on random-graphs where each user has a small, fixed number of connections to others in the graph. These topologies have proven not to scale well.

We are investigating the use of partial hypercubes as the topology for peer-to-peer networks useful in genealogical research. We assume that peers are continuously connected and that they are not anonymous. We call our implementation of these networks, hyperwebs. Our simulation research has shown that hyperwebs provide scalable topologies well-suited to the type of communication needed for effective genealogical research.

In our presentation we will describe our implementation of hypercubes as a topology for peer-to-peer networking. Research in hypercubes was originally motivated by the need for efficient topologies in parallel computation. Most of the work focused on perfect hypercubes with 2^n nodes. Interest in using hypercubes for parallel computation has waned but the scalability properties of hypercubes makes them attractive for use in peer-to-peer topologies. To be able to use them we need a structure that preserves the property that every node is connected to at least $log(n)$ other nodes but does not require the hypercube to be perfect. We describe our solution to this problem which is a second-order partial hypercube. We chose a second-order solution so as to facilitate simultaneous additions and deletions to and from a hypercube. We will describe our add and delete algorithms and how they can be implemented efficiently.

The hypercube is only useful if we can broadcast information efficiently. We will describe an $O(log(n))$ minimal cost broadcast algorithm that we have implemented. If hypercubes are to be used for distributed databases, they must provide an efficient search mechanism. Current peer-to-peer topologies do not. We describe an indexing structure that distributes index information throughout the hyperweb. With it we can determine in $O(1)$ time which node in the hyperweb contains the index information of interest, and, in $O(log(n))$ time we can retrieve the information. Thus, looking up indexes is $O(log(n))$.

Last of all we will show a prototype hypercube-based network that demonstrates some of the previously described features. The prototype is that of a simple research assistant for genealogical research.