

# P2P: A Natural Way to Communicate

Jason A. Wood

May 3, 2001

*Copyright (c) 2001 by Jason A. Wood. Permission is granted to copy, distribute and/or modify this document under the terms of the Open Publication License <http://www.opencontent.org/openpub/>, version 1.0 or later.*

## **Abstract**

Peer to Peer computing has received a lot of negative press recently. These include the big names such as Gnutella and Napster. Peer to peer (P2P), however, has many more important merits and consequences than circumscribing copyright. This paper explores briefly current P2P technology, as well as a suggested framework for P2P in the future.

## **Introduction**

Peer to Peer (P2P) is one of the many '2' buzzwords in the IT sphere today. The challenge of B2B, B2C, and now P2P is to make it easier for communication between two entities in an efficient manor. Currently much money is made from the spread of information from its source to its destination. A musical artist for instance records their music at a studio and then the recording company and retailers make huge sums of money in the spreading of the music, or more generically information. This same scheme is found in the news industry, commerce, and government. Each has a rigid hierarchical structure that links entity with entity, person with person.

What P2P technology allows is for people to cut out the intermediary and communicate directly with source and with less effort than traditional methods provide. What will make this possible is for people to create their own virtual middlemen or agents. The benefit of such a new system will be each persons own desires will be met in a way that is agreeable between them and the provider without external influence. If a band wishes, therefore, to give a favorite fan the music for free and make someone else pay for the same music, that is easily made possible.

## **Present View**

Right now P2P is in its infancy and many who fear its effects on the status quo would like to wish it a quick death.[1] The major problem with P2P in most of its current implementations is lack of scalability. The systems promote a decentralized view, but rely on a single central directory to provide initial connections. The Napster protocol defines the role of a server, which provides a directory of what mp3s each user has available. Similarly systems such as AOL

Instant Messenger follow the same method for bringing users together through their Oscar server.[2] These systems are for the most part a close cousin to the client-server architecture.

## **Future of P2P**

P2P will continue for the foreseeable future use the client-server architecture. The difference is the idea of a centralized server will disappear and each computer will be both a client and a server. There are many different ways to supporting this model. One of the earliest methods is SMB/CIFS, which is a Microsoft sponsored protocol for file sharing across a network. In this network any computer can be the directory for a particular domain. Then each computer on the network registers itself with the directory or NetBOIS Name Server. One person who wants to retrieve files from another person they can do a lookup based upon the registered name of the other person's computer on the network through the NetBOIS server, resolve their IP address, and then begin a P2P session.[3]

The P2P system that I envision, however, will be similar to the X.500 protocol for name resolution using a DNS like protocol for directory services. With current methods if an organizations connection to the web breaks, so does their interoffice communication structure if they rely on AOL's servers or any other external provider. In the system I propose, each organization, or each person has their own directory. This directory contains links between each person in a hierarchical structure based on their context. So if one person within the company wishes to speak to another person in the company that connection doesn't rely on anything external to the company, since their context matches that of the other person.

As an illustrative example, Kathy wishes to speak to Jim. Kathy is identified as "uid=kathy, ou=Administration, dc=somecompany, dc=com" and Jim is identified as "uid=jim, ou=Marketing, dc=somecompany, dc=com." Kathy's P2P application only knows about Kathy's PC and her identifier or distinguished name (dn). To contact Jim, it does a search for Jim using the local directory server. Once it locates Jim's entry it can then send information back to Kathy's PC, so she can create a P2P connection with Jim. Connections outside of the organization are very similar, but they require a lookup through the next level up in the hierarchy, in this case the directory of .com's.

Another important addition to P2P will be "A2A", that is Agent to Agent. In the future people will be able to create agents, which act on their behalf and closely follow the same behavior the individual would exhibit. This will allow people to create what would have been middlemen, but in such a way that suits them, not outside interests or that of the middleman. These agents would then interact in an electronic manor to conduct business and communications. This provides an asynchronous communication structure for industry, commerce, and government that is highly individualised. My agents would be my representatives in an electronic congress and my marketing team to sell my services.

## **Conclusions**

P2P as long as distributed networks are available will become as common as their present counterparts. Interactions between people and organizations will continue in much the same way as they are now, but through electronic means that more closely resemble the non-technological interactions common today. I hope this will result in a society that

is tailored more towards the individual than impersonal methods currently required for efficient and safe society.

## References

- [1] Lee Gomes. "Next on record industry's hit list–Napster clones." WSJ Interactive Edition. February 23, 2001. <http://www.zdnet.com/zdnn/stories/news/0,4586,2689187,00.html>. Accessed April 30, 2001.
- [2] Adam Fritzler. "AIM/Oscar Protocol Specification." Apr. 8, 2000. <http://www.zigamorph.net/faim/protocol/section3.html>. Accessed April 30, 2001.
- [3] Robert Eckstein, et. al. Using Samba. O'Reilly and Associates: Nov. 1999. [http://www.oreilly.com/catalog/samba/chapter/book/ch01\\_03.html](http://www.oreilly.com/catalog/samba/chapter/book/ch01_03.html). Accessed April 30, 2001.