



What's your line?

Create a T-shirt slogan any geek would love and win a \$50 shopping spree at ThinkGeek—and your slogan on a shirt



Published on [The O'Reilly Network](http://www.oreillynet.com/) (<http://www.oreillynet.com/>)
<http://www.oreillynet.com/pub/a/p2p/2000/11/24/shirky1-whatisp2p.html>
 See [this](#) if you're having trouble printing code examples

What Is P2P... And What Isn't ?

by [Clay Shirky](#)
 11/24/2000

After a year or so of attempting to describe the revolution in file sharing and related technologies, we have finally settled on a label for what's happening: peer-to-peer.

Somehow, though, this label hasn't clarified things. Taken literally, servers talking to one another are peer-to-peer. The game Doom is peer-to-peer. There are even people applying the label to e-mail and telephones. Meanwhile, Napster, which jump-started the conversation, is not peer-to-peer in the strictest sense, because it uses a centralized server to store pointers and resolve addresses.

If we treat peer-to-peer as a literal definition for what's happening, then we have a phrase that describes Doom but not Napster, and suggests that Alexander Graham Bell was a peer-to-peer engineer but Shawn Fanning is not.

This literal approach to peer-to-peer is plainly not helping us understand what makes P2P important. Merely having computers act as peers on the Internet is hardly novel, so the fact of peer-to-peer architecture can't be the explanation for the recent changes in Internet use.

What *has* changed is what the nodes of these P2P systems are -- Internet-connected PCs, which had been formerly relegated to being nothing but clients -- and where these nodes are -- at the edges of the Internet, cut off from the DNS system because they have no fixed IP address.

Resource-centric addressing for unstable environments

P2P is a class of applications that takes advantage of resources -- storage, cycles, content, human presence -- available at the edges of the Internet. Because accessing these decentralized resources means operating in an environment of unstable connectivity and unpredictable IP addresses, P2P nodes must operate outside the DNS system and have significant or total autonomy from central servers.

That's it. That's what makes P2P distinctive.

Note that this isn't what makes P2P important. It's not the problem designers of P2P systems set out to solve -- they wanted to create ways of aggregating cycles, or sharing files, or chatting. But it's a problem they all had to solve to get where they wanted to go.

Related Articles:

[What's On Freenet?](#)

[Free Radical: Ian Clark has Big Plans for the Internet](#)

[How Ray Ozzie Got His Groove Back](#)

[A Directory of Peer-to-Peer Projects](#)

[Open Source Roundtable: Free Riding on Gnutella](#) 

[O'Reilly's Peer-to-Peer](#)

What makes Napster and ICQ and Popular Power and Freenet and AIMster and Groove similar is that they are all leveraging previously unused resources, by tolerating and even working with the variable connectivity of the hundreds of millions of devices that have been connected to the edges of the Internet in the last few years.

[Conference](#)

[More from the P2P DevCenter](#) ▶
[Previous Features](#) ▶

One could argue that the need for P2P designers to solve connectivity problems is little more than an accident of history, but improving the way computers connect to one another was the rationale behind IP addresses, and before that DNS, and before that TCP, and before that the net itself. The internet is made of such frozen accidents.

P2P is as P2P does

Up until 1994, the whole Internet had one model of connectivity. Machines were assumed to be always on, always connected, and assigned permanent IP addresses. The DNS system was designed for this environment, where a change in IP address was assumed to be abnormal and rare, and could take days to propagate through the system.

With the invention of Mosaic, another model began to spread. To run a Web browser, a PC needed to be connected to the Internet over a modem, with its own IP address. This created a second class of connectivity, because PCs would enter and leave the network cloud frequently and unpredictably.

Furthermore, because there were not enough IP addresses available to handle the sudden demand caused by Mosaic, ISPs began to assign IP addresses dynamically, giving each PC a different, possibly masked, IP address with each new session. This instability prevented PCs from having DNS entries, and therefore prevented PC users from hosting any data or net-facing applications locally.

For a few years, treating PCs as dumb but expensive clients worked well. PCs had never been designed to be part of the fabric of the Internet, and in the early days of the Web, the toy hardware and operating systems of the average PC made it an adequate life-support system for a browser, but good for little else.

Over time, though, as hardware and software improved, the unused resources that existed behind this veil of second-class connectivity started to look like something worth getting at. At a conservative estimate, the world's Net-connected PCs presently host an aggregate ten billion Mhz of processing power and ten thousand terabytes of storage, assuming only 100 million PCs among the net's 300 million users, and only a 100 Mhz chip and 100 Mb drive on the average PC.

The veil is drawn back

The launch of ICQ in 1996 marked the first time those intermittently connected PCs became directly addressable by average users. Faced with the challenge of establishing portable presence, ICQ bypassed DNS in favor of creating its own directory of protocol-specific addresses that could update IP addresses in real time, a trick followed by Groove, Napster, and NetMeeting as well. (Not all P2P systems use this trick. Gnutella and Freenet, for example, bypass DNS the old-fashioned way, by relying on numeric IP addresses. Popular Power and SETI@Home bypass it by giving the nodes scheduled times to contact fixed addresses, thus delivering their current IP address at the time of the connection.)

Whois counts 23 million domain names, built up in the 16 years since the inception of IP addresses in 1984. Napster alone has created more than 23 million non-DNS addresses in 16 months, and when you

add in all the non-DNS Instant Messaging addresses, the number of P2P addresses designed to reach dynamic IPs tops 200 million. Even if you assume that the average DNS host has 10 additional addresses of the form foo.host.com, the total number of P2P addresses now equals the total number of DNS addresses after only 4 years, and is growing faster than the DNS universe today.

As new kinds of Net-connected devices like wireless PDAs and digital video recorders like TiVo and Replay proliferate, they will doubtless become an important part of the Internet as well, but for now PCs make up the enormous preponderance of these untapped resources. PCs are the dark matter of the Internet, and their underused resources are fueling P2P.

Litmus tests

If you're looking for a litmus test for P2P, this is it: 1) Does it treat variable connectivity and temporary network addresses as the norm, and 2) does it give the nodes at the edges of the network significant autonomy?

If the answer to both of those questions is yes, the application is P2P. If the answer to either question is no, it's not P2P.

Another way to examine this distinction is to think about ownership. It is less about "Can the nodes speak to one another?" and more about "Who owns the hardware that the service runs on?" The huge preponderance of the hardware that makes Yahoo work is owned by Yahoo and managed in Santa Clara. The huge preponderance of the hardware that makes Napster work is owned by Napster users and managed on tens of millions of individual desktops. P2P is a way of decentralizing not just features, but costs and administration as well.

Real solutions to real problems

We have unpredictable IP addresses because there weren't enough to go around when the web happened. It's tempting to think that when enough new IP addresses are created, though, the old "One Device/One Address" regime will be restored, and the Net will return to its pre-P2P architecture.

This won't happen, because no matter how many new IP addresses there are, P2P systems often create addresses for things that aren't machines. Freenet and MojoNation create addresses for content intentionally spread across multiple computers. AIM and ICQ create names which refer to human beings and not machines. P2P is designed to handle unpredictability, and nothing is more unpredictable than the humans who use the network. As the Net becomes more human-centered, the need for addressing schemes that tolerate and even expect temporary and unstable patterns of use will grow.

Who's who?

Napster is P2P, because the addresses of Napster nodes bypass the DNS system, and because once the Napster server resolves the IP addresses of the PCs hosting a particular song, it shifts control of the file transfers to the nodes. Furthermore, the ability of the Napster nodes to host the songs without central intervention lets Napster users get access to several terabytes of storage and bandwidth at no additional cost.

However, Intel's "server peer-to-peer" is not P2P, because servers have always been peers. Their fixed IP addresses and permanent connections present no new problems, and calling what they already do "peer-to-peer" presents no new solutions.

ICQ and Jabber are P2P, because not only do they devolve connection management to the individual nodes once they resolve the addresses, they violate the machine-centric worldview encoded in the DNS system. Your address has nothing to do with the DNS systems, or even with a particular machine, except temporarily -- your chat address travels with you. Furthermore, by mapping "presence" -- whether you are at your computer at any given moment in time -- chat turns the old idea of permanent connectivity and IP addresses on its head. Chat is an important protocol *because* of the transience of the connectivity.

E-mail, which treats variable connectivity as the norm, is nevertheless not P2P, because your address is not machine independent. If you drop AOL in favor of another ISP, your AOL e-mail address disappears as well, because it hangs off DNS. Interestingly, in the early days of the Internet, there was a suggestion to make the part of the e-mail address before the @ globally unique, linking e-mail to a person rather than to a person@machine. That would have been P2P in the current sense, but it was rejected in favor of a machine-centric view of the internet.

Popular Power is P2P, because the distributed clients that contact the server need no fixed IP address and have a high degree of autonomy in performing and reporting their calculations, and can even be offline for long stretches while still doing work for the Popular Power network.

Dynamic DNS is not P2P, because it tries to retrofit PCs into the traditional DNS system, and so on.

This list of resources that current P2P systems take advantage of -- storage, cycles, content, presence -- is not necessarily complete. If there were some application that needed 30,000 separate video cards, or microphones, or speakers, a P2P system could be designed that used those resources as well.

P2P is a horseless carriage

Whenever something new seems to be happening on the Internet, there is a push to define it, and as with the "horseless" carriage or the "compact" disc, new technologies are often labelled according to some simple difference from what came before -- horsedrawn carriages, non-compact records.

Calling this new class of applications peer-to-peer emphasizes their difference from the dominant client/server model. However, like the horselessness of the carriage or the compactness of the disc, the "peeriness" of P2P is more a label than a definition.

As we've learned from the history of the Internet, adoption is a better predictor of software longevity than perfection is, and as the P2P movement matures, users will not adopt applications that embrace decentralization for decentralization's sake. Instead, they will adopt those applications that use just enough decentralization, in just the right way, to create novel functions or improve existing ones.

Clay Shirky writes about the Internet and teaches at NYU's Interactive Telecommunications Program. He publishes a mailing list on Networks, Economics, and Culture at shirky.com/nec.html.

Related Articles:

[What's On Freenet?](#)

[Free Radical: Ian Clark has Big Plans for the Internet](#)

[A Directory of Peer-to-Peer Projects](#)

[How Ray Ozzie Got His Groove Back](#)

[Open Source Roundtable: Free Riding on Gnutella](#) 

[O'Reilly's Peer-to-Peer Conference](#)

Discuss this article in the O'Reilly Network [General Forum](#).

Return to the [P2P DevCenter](#).

oreillynet.com Copyright © 2003 O'Reilly & Associates, Inc.