

IEEE DS ONLINE EXCLUSIVE CONTENT

From the Editors: Peer-to-Peer Community

Looking beyond the Legacy of Napster and Gnutella

Kiran Nagaraja • NEC Laboratories

Sami Rollins • Mount Holyoke College

Mujtaba Khambatti • Microsoft

Trace back to 1999: that was the year Napster took the music world by storm. The application brought together millions of music enthusiasts eager to share and download each other's song collections. The ease with which two "peers" could share a file was refreshing. At its peak in early 2000, this rage had resulted in a network of about 60 million Napster users. It was also around this time that the phrase *peer to peer* (P2P) came to be associated with systems such as Napster. The phenomenon not only stirred the social sphere but also created a buzz in the academic and industrial worlds. Anyone who knew anything about distributed systems was asking, is P2P computing really a new paradigm?

The short answer was no. Usenet news and email systems (those based on the Simple Mail Transfer Protocol, or SMTP) are examples of systems, invented as far back as the 1980s, that used the same decentralization concept that is characteristic of P2P. The detailed answer, however, is that the field of P2P has made enough novel contributions in data structures, algorithms, and mechanisms over the past five years to justify its distinction from distributed systems in general. However, despite this innovation, a summary look at the applications that have emerged from this field tells a mostly monotonic story.

The legacy

Owing to the popularity of Napster and its successors, including Gnutella, Kazaa, Morpheus, and E-Donkey, file sharing has become by far the killer P2P application. Its popularity almost eclipses other P2P applications. Moreover, much P2P research has targeted the challenges facing these popular file-sharing networks. Gnutella addressed Napster's shortfall of complete decentralization, but its unstructured nature raised concerns over its search mechanism's efficiency and scalability. In 2001, P2P applications started to use *superpeers* (a set of more powerful nodes in a heterogeneous network) to transform the existing flat topology of these networks into a hierarchical one. Superpeers are considered faster and more reliable than normal peers and take on server-like responsibilities. For example, in the case of file sharing, a superpeer builds an index of the files shared by its "client" peers and participates in the search protocol on their behalf. This improves scalability by limiting the flood of search traffic.

Simultaneously, various projects such as Chord, Pastry, Tapestry, and CAN (content-addressable network) proposed using distributed hash tables to locate objects in a network. DHTs offered a structured way to build an overlay that would efficiently support the mapping of object identifiers to their locations. Even though such core contributions had their root in solving the decentralized-search problem of file-sharing systems, it became widely accepted by the research community that the core operation in most P2P systems was efficiently locating data objects. Apart from services for content location and query routing, researchers also actively worked on providing content security and availability and establishing authentication, trust, and the reputation of individual peers. In summary, advances along these directions have helped establish a firm base to explore a complex set of applications beyond file sharing.

Other domains

The hype raised enough curiosity among researchers and developers to apply P2P mechanisms to domains other than file sharing. Some early explorations were in distributed storage, content distribution, communication and collaboration, and to an extent even decentralized gaming. (We purposely exclude distributed-computing applications such as grid computing. Like P2P, these systems employ decentralization and incremental scalability, but their usage is often managed either at a single location or at multiple ones in a federated manner.) Reduced management costs per entity, absence of central points of failure, incremental scalability, a potentially larger resource pool, reduced individual scrutiny, and potentially higher fault resilience were the primary advantages over centralized solutions. However, except for a few real products such as Groove Network's Virtual Office for enterprise-wide collaboration, most of these efforts remain academic.

In fact, only the use of P2P networks for distributed storage and related services can match the level of interest file sharing has generated. Oceanstore, PAST (*P2P archival storage*), Pastiche, PeerStore, and CFS (cooperative file system) are among the notable academic projects that propose using P2P networks for universal access to or the archiving of content. Content distribution also received interest. Networks such as Akamai and BitTorrent optimize network usage for fast, efficient download of content by end hosts. Essentially, despite all the advances in developing core services, the application of P2P has remained well within the domain of content sharing. This field has yet to see revolutionary applications beyond what Napster demonstrated.

The revolution is coming?

P2P has democratized the way people use computing. It links people irrespective of their location and affiliation and raises numerous possibilities of interaction and collaboration not only in social activities but also on a commercial level. First, it presents an opportunity for sharing digital content, as we've already described. It also presents an opportunity for sharing computing and other resources. Although some applications such as SETI@Home harness idle computing cycles from a pool of Internet hosts, none let individuals gather and use such resources by mutual consent.

Second, it enables unbounded social interaction and collaboration without depending on centralized rendezvous points. Applications such as communication, conferencing, network gaming, voting, opinion polling, and synchronized viewing of video streams can leverage this P2P feature. Existing applications such as NetMeeting and some features in today's instant-messaging systems give us glimpses of what's possible. A few P2P applications support collaboration between people in an enterprise, but they're specialized to deal with data and processes in a work setting.

Finally, the ultimate test for P2P's success will be its ability to support applications of commercial value over the network. Establishing marketplaces and auctions in a P2P setting would create the equivalent of today's Amazon and eBay without the negatives associated with centralization.

Current research is certainly targeting a broader P2P application space. Several proposals exist to build more complex applications using the existing base of efficient, scalable core services. Examples include a digital library,¹ a complex, massively multiplayer online game that uses Chord for object location,² and a commercial content distribution network that assumes the presence of neutral superpeers to perform bookkeeping for downloads between the provider and consumer.³ Research is also ongoing to construct network-wide infrastructures, such as a P2P-based Domain Name System⁴ and a spam-filtering service⁵ that can improve overall performance of legacy and P2P applications alike.

However, for all this to translate into practice, application developers must be able to benefit from the solid

platform the research community has established. To see if this aspect is appreciated within the research community, consider the breakdown of research topics (see figure 1) addressed by papers presented over the last four years at the International Workshop on Peer-to-Peer Systems, a prominent P2P workshop. IPTPS encourages submissions to discuss P2P's state-of-the-art and to identify key research challenges. Deservedly, core services have received the most attention over the years, and we expect this trend to continue. Applications, however, have been neglected. More disappointing is the small fraction of submissions that address application developers' concerns—represented by Standards in figure 1. Challenges that affect developers, and which the community has not adequately addressed, include

- defining the appropriate set of interfaces to be exported by core services to ease application development (apart from performance, reliability, security, and so on), and
- determining the appropriate data structure, algorithm, or mechanism (and the parameters involved) for an application, given a set of requirements and environmental preconditions.

To repeat a point that one IPTPS paper made, all the research done will receive neither feedback nor validation unless there's an active set of clients for the technology. This appears to be a major problem. Efforts such as JXTA, which invite the community to be involved in open source development of P2P specifications, standards, and technology, are steps in the right direction. The systems community involved in P2P research must step up to this challenge as well.

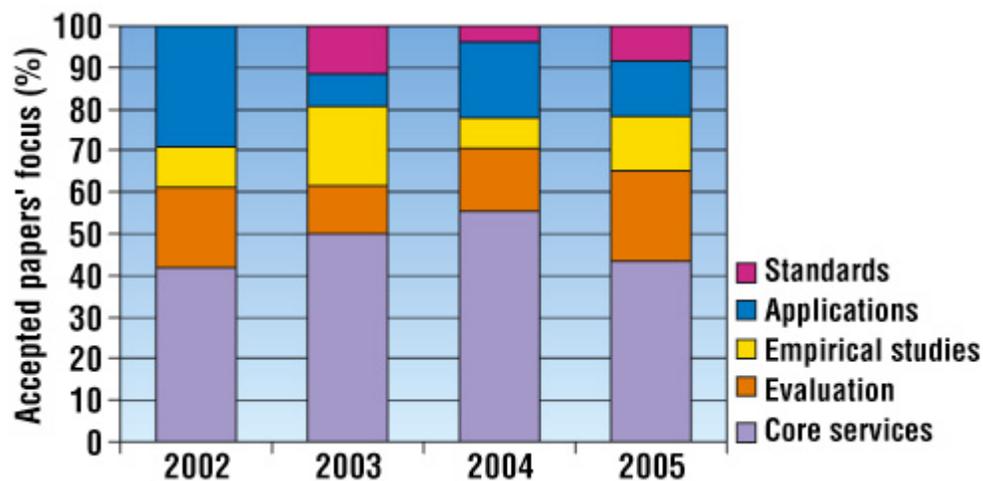


Figure 1. P2P researchers' focus as evident from papers accepted at the International Workshop on Peer-to-Peer Systems from 2002 to 2005.

The field of P2P has seen significant advances over the last five years. The set of core structures, algorithms, and mechanisms developed in this community have helped address various issues, including scale, performance, availability, security, and trust within these networks. However, despite these advances, the field has yet to realize its full potential in the application domain. It's time to move beyond file-sharing applications and fill this void.

References

1. J. Stribling et al., "OverCite: A Cooperative Digital Research Library," *Proc. 4th Int'l Workshop Peer-to-Peer Systems (IPTPS 05)*, LNCS 3640, Springer, 2005, pp. 69–79. ([pdf](#))

2. S. Douglas et al., "Enabling Massively Multi-Player Online Gaming Applications on a P2P Architecture," *Proc. Int'l Conf. Information and Automation*, IEEE Press, 2005, pp. 7-12.
3. A. Wierzbicki and K. Goworek, "Peer-to-Peer Direct Sales," *Proc. 5th Int'l Conf. Peer-to-Peer Computing (P2P 05)*, IEEE CS Press, 2005, pp. 106-113.
4. K. Park et al., "CoDNS: Improving DNS Performance and Reliability via Cooperative Lookups," *Proc. 6th Symp. Operating Systems Design and Implementation (OSDI 04)*, Usenix, 2004, pp. 199-214.
5. F. Zhou et al., "Approximate Object Location and Spam Filtering on Peer-to-Peer Systems," *Proc. ACM/IFIP/Usenix Int'l Middleware Conf. (Middleware 03)*, LNCS 2672, Springer, 2003, pp. 1-20.



Kiran Nagaraja is a research staff member with the Systems LSI Group at NEC Laboratories. Contact him at kiran@nec-labs.com.



Sami Rollins is an assistant professor of computer science at Mount Holyoke College. Contact her at srollins@mtholyoke.edu.



Mujtaba Khambatti is a program manager in Microsoft's Windows Reliability Team. Contact him at mujtabak@microsoft.com.

Related Links



[DS Online's Peer-to-Peer Community](#)



[5th IEEE International Conference on Peer-to-Peer Computing \(P2P 2005\)](#)



["Is Deviant Behavior the Norm on P2P File-Sharing Networks?"](#)



["Approaches to P2P Internet Application Development"](#)

Cite this article:

Kiran Nagaraja, Sami Rollins, and Mujtaba Khambatti, "Looking beyond the Legacy of Napster and Gnutella," *IEEE Distributed Systems Online*, vol. 7, no. 3, 2006, art. no. 0306-o3005.



dsonline.computer.org

 PRINT

 CLOSE