Peer-to-Peer Networking and Applications: Synopsis and Research Directions

John F. Buford and Heather Yu

Abstract Peer-to-peer computing and networking are important developments for large-scale distributed systems design and the evolution of Internet architecture. Widely used applications have demonstrated their feasibility and economic potential for services involving millions of users. A great deal of research has followed to formalize and improve on the empirical results. This introductory chapter surveys the key results of the field, introduces terminology, and identifies open issues which are likely to be important research directions.

1 Introduction

1.1 Significance and Emergence

Several important desktop computing applications have emerged in recent years that use an Internet-scale decentralized architecture to simultaneously connect millions of users to share content, form social groups and communicate with their contacts. These applications are classified as peer-to-peer because of the elimination of servers to mediate between end systems on which the applications run, and their network behavior is described as an overlay network because the peer protocols form a virtualized network over the physical network.

While peer-to-peer (P2P) applications have had a rapid ascent and wide impact, in the future P2P overlays are likely to enable important new applications following from these technology trends:

John F. Buford
Avaya Labs Research, Basking Ridge, NJ 07920, USA, e-mail: buford@samrg.org
Heather Yu
Huawei Technologies, Bridgewater, NJ, USA, e-mail: heathery@ieee.org

X. Shen et al. (eds.), *Handbook of Peer-to-Peer Networking,*
DOI 10.1007/978-0-387-09751-0_1,

– Continued improvements in the fidelity of the consumer entertainment experience and network and computing capacity of the associated entertainment devices
– The development of dense and ubiquitous sensing grids with real-time data collection of all types of phenomena
– The wide deployment of broadband wireless networks (WiMax, 802.11n, UWB, LTE)
– The proliferation of mobile smart phones and other broadband-enabled mobile devices
– The use of personal networks, body-area networks, and vehicle networks, to connect both real-time sensors and embedded computing devices.

The wide adoption of these technologies will enable high-fidelity and pervasive information collection, content publishing and distribution, and sharing of environmental and personal real-time sensed data and information on a global scale. The benefits of this include increased awareness of one's personal environment, more precise context-awareness in interactions with others, and enhanced situation-awareness for applications ranging from immersive entertainment and recreation, environment management, homeland security, and disaster recovery. Peer-to-peer overlays are an important component of this future vision, due to their high scalability, flexibility for different types of applications, and low barrier of entry. The evolution of contemporary P2P overlays to enable this future vision is an important research direction.

The use of application layer protocols to form overlays to deliver Internet services has a long history (Table 1). However until relatively recently, these types of overlays used specifically designed protocols, and were used to interconnect infrastructure servers rather than end systems. In addition, the address space of the overlay was typically not virtualized, and dealing with churn was not a primary design point. Nevertheless, such service overlays continue to be an important part of the Internet architecture [1–3], and there is growing interest in using the end-to-end and resource virtualization capabilities of overlays in the evolution of the Internet architecture. Example research efforts in this direction include SpoVNet [4] and SATO [5].

Table 1 Specialized overlay networks for internet services

Type	Example	First use or definition
Email	SMTP	1970s
Internet news	NNTP	1986
Multicast	MBone	1992
Web caching	Internet cache protocol	1995
Content delivery network	Akamai	1999
Anonymous communication	FreeNet	1999
Application layer multicast	Narada	2000
Routing	RON	2001

The well-known popularization of P2P file sharing systems beginning with the hybrid Napster and followed by other content dowloading P2P systems such as Gnutella, FastTrack, KaZaa, and BitTorrent invigorated the interest of the research community to develop solutions to the perceived deficiences of these systems. The subsequent availability of P2PTV and VoP2P applications discussed in the next section were further catalysts for research in real-time media streaming over P2P overlay networks.

Notable underpinnings of the research in improving P2P overlays was the early work on distributed hash tables by Devine [6] as well as Litwin et al. [7, 8]. Plaxton, Rajaraman, and Richa (PRR) [9] presented the first algorithms for distributed object location and routing, using a suffix forwarding scheme. PRR was the basis for subsequent influential designs such as Tapestry and Pastry. Karger, et al. [10] formalized consistent hashing which is the basis for many DHT designs.

1.2 Key Applications

The first widely used file sharing system, Napster, featured a hybrid architecture in which the directory was stored on a server, but peers directly transferred files between them. Napster became the first legal test case for file sharing of licensed content, and was subsequently forced to change to protect such content. A number of peer-to-peer file sharing systems were developed (Table 2) to avoid the legal challenges faced by Napster. The majority of these second-generation file sharing systems were based on unstructured overlays. While these systems had no mechanisms for protecting the rights of content owners, in some cases the P2P application developers obtained revenue by either selling their applications or by embedding

Table 2 Example file sharing applications

Client application(s)	**Protocol**	**Description**
KaZaA grokster imesh	FastTrack	Proprietary unstructured overlay with encrypted protocol, high capacity peers become superpeers; features connection shuffling
Limewire	Gnutella	Superpeer unstructured overlay with flooding query propagation
eDonkey	Overnet	Structured overlay based on Kademlia
eMule	Kad	Structured overlay based on Kademlia
BitTorrent	BitTorrent	An unstructured overlay used for distributing large files in pieces using mutual distribution of the pieces between a set of peers called a swarm. Uses a server to store the torrent and another server called a tracker to identify the swarm members

spyware in the clients. Recent research studies of P2P file sharing systems include [11–16].

Several new ventures such as QTrax, SpiralFrog, and TurnItUp have proposed incorporating DRM in to the file sharing applications or ad-based revenue models in which advertisements are delivered during media playback.

The founders of KaZaA subsequently launched the first widely used voice-over P2P (VoP2P) application, Skype. Currently Skype connects around 15 M concurrent users and provides a variety of services including P2P voice and video calls, voice calls to PSTN endpoints, presence, and instant messaging. Like KaZaA, the Skype protocol is encrypted and the definition of the protocol has not been released. Some studies have shown that Skype uses a superpeer model, and the superpeers support NAT traversal for connecting peers behind NATs. In addition, superpeers also act as media relays. Recent research studies of Skype include [17–21, 101].

In contrast to file sharing systems which exhibit the free rider behavior, in P2P telephony users are motivated to stay connected both to be able to receive calls and view the current status of their buddies. Long application lifetimes mean a low churn rate, which makes the operation of the overlay more stable. Experimental studies of Skype have shown a significantly higher node lifetime compared to P2P file sharing sytems.

The distribution of streaming video referred to as P2PTV has also become an important application of P2P. Various models are used, including torrent-style distribution, application layer multicasting, and hybrid CDNs (content delivery networks). Example PPTV applications include Babelgum, Joost, PPLive, PPStream, SopCast, TVants, TVUPlayer, Veoh TV, and Zattoo. P2PTV is expected to play an important role in future IPTV deployments. A summary of P2PTV related research is discussed later in this chapter.

1.3 Definition and Properties of P2P Systems

Peer-to-peer systems have been defined in many papers. Here are two definitions that cover the concepts of resource sharing, self-organization, decentralization, and interconnection:

> "A distributed network architecture may be called a peer-to-peer network, if the participants share a part of their own hardware resources (processing power, storage capacity, network link capacity, printers). These shared resources are necessary to provide the Service and content offered by the network (e.g. file sharing or shared workspaces for collaboration). They are accessible by other peers."[22]
>
> "Peer-to-peer systems are distributed systems consisting of interconnected nodes able to self-organize into network topologies with the purpose of sharing resources such as content, CPU cycles, storage and bandwidth, capable of adapting to failures and accommodating transient populations of nodes while maintaining acceptable connectivity and performance, without requiring the intermediation or support of a global centralized server or authority." [23]

We have also defined an overlay network [26]:

> "An application layer virtual or logical network in which end points are addressable and that provides connectivity, routing, and messaging between end points. Overlay networks are frequently used as a substrate for deploying new network services, or for providing a routing topology not available from the underlying physical network. Many peer-to-peer systems are overlay networks that run on top of the Internet."

The following properties are characteristics found in most P2P systems.

Resource sharing each peer contributes system resources to the operation of the P2P system. Ideally this resource sharing is proportional to the peer's use of the P2P system, but many systems suffer from the free rider problem.

Networked all nodes are interconnected with other nodes in the P2P system, and the full set of nodes are members of a connected graph. When the graph is no longer connected, the overlay is said to be partitioned.

Decentralization the behavior of the P2P system is determined by the collective actions of peer nodes, and there is no central control point. Some systems however secure the P2P system using a central login server. The ability to manage the overlay [24] and monetize its operation may require centralized elements.

Symmetry nodes assume equal roles in the operation of the P2P system. In many designs this property is relaxed by the use of special peer roles such as super peers or relay peers.

Autonomy participation of the peer in the P2P system is determined locally, and there is no single administrative context for the P2P system.

Self-organization the organization of the P2P system increases over time using local knowledge and local operations at each peer, and no peer dominates the system. Biskupski, Dowling, and Sacha [25] argue that existing P2P systems do not exhibit most of the properties of self-organization.

Scalable This is a pre-requisite of operating P2P systems with millions of simultaneous nodes, and means that the resources used at each peer exhibit a growth rate as a function of overlay size that is less than linear. It also means that the response time doesn't grow more than linearly as a function of overlay size.

Stability Within a maximum churn rate, the P2P system should be stable, i.e., it should maintain its connected graph and be able to route deterministically within a practical hop-count bounds.

1.4 Business Models

P2P file sharing applications have been monetized by their operators by sales of the P2P client software or by embedding spyware into the application. Content licensing has not been as successful to date. The leading VoP2P application, Skype, provides basic P2P telephony for free, and receives revenue for add-on services such

as voice mail or peer-to-PSTN calls. The primary model for P2PTV operators is similar to existing cable and broadcast TV – embedded advertising with measurable viewership.

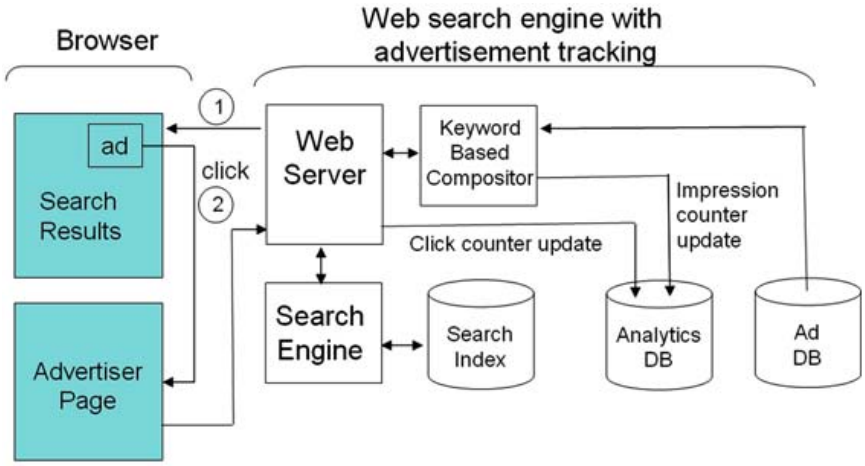
The advertising mechanisms used in web search (impressions, cost-per-click, and placements) are more difficult to implement in P2P applications since verification can not rely on a centralized collection point. Figure 1a shows a simplified model for impression and click counting in web search. The search results are produced by the search engine, and in parallel the search keywords and other criteria are used to select ads which will be composed with the search results, resulting in an impression for each displayed ad. Each time an impression occurs, a counter is updated in the advertisement analytics database. If the user clicks on the ad, the embedded url leads to the advertiser's web page. A click counter must be updated in the advertisement analytics databse. Two ways to obtain this are to indirectly forward the ad url via a specific search engine web server (e.g., www.searchengine.com?advertiser.com), or to embed a script in the advertiser's web page which invokes the search engine's url when loaded. The advertiser can also use third party services to track web site hits with analytics reports produced by the search vendor.

In the P2P case, assume ads are selected for display on the P2P application user interface by association with searches initiated through the application user interface. (Of course, the P2P application developer could sell banner ad space on the P2P applications that would point to advertiser's web sites, but these would not be specifically targeted according to the user's activities or application use). Figure 1b shows the basic flow. A search request is propagated through the P2P network, returning one or more search results from various peers. Ads which match the search criteria are returned with the search results. There are several ways this could be done, using the P2P network or using a separate index, and monetization could be obtained by both the overlay operator and the operators of the peers that return the search results. For this discussion, serving advertisements through peers raises the question of how to validate impression counts maintained at the peers, as well as click-through counts at the peer application.

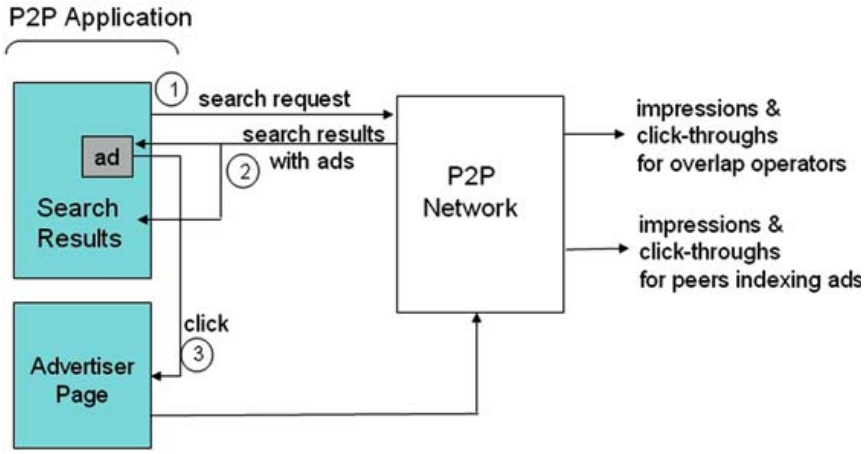
1.5 Technology Drivers

The P2P value proposition [26] "... for the user is to exchange excess computational, storage, and network resources for something else of value to the user, such as access to other resources, services, content, or participation in a social network." The rapid gains in computer capacity and wide adoption of broadband access have thus fueled the growth of P2P applications. As these trends are expected to continue, we raise the following questions:

– Given the limited ability of search engines to index all of the web, can P2P search compete with or augment web search, and at what scale and cost?



(a) Click-through and impression tracking in web search



(b) Click-through and impression tracking in P2P networks

Fig. 1 Click through and impression tracking

- Currently the largest online peer populations are in the range of 10–20 M. Can P2P networks support continued growth to say 10x or 100x peer population increase? What are the limits?
- Given the emergence of broadband wireless and the likely domination of peer population by mobile wireless devices versus fixed desktop devices, how will the stability and operation of P2P networks be effected?
- Will HDTV and high-definition video lead to new P2P applications?
- Will deployment of large-scale sensor grids create new applications for P2P networks, and what are appropriate architectures for interconnecting such networks in a global overlay?

1.6 Structure of the Chapter

The remainder of the chapter is organized as follows. First we survey overlay design, and provide a taxonomy for understanding the many different overlays that have been proposed. This survey includes unstructured, structured, hierarchical, service, semantic, and sensor overlays. The next section summarizes results on overlay dynamics, including mobility and overlays for MANETs, and variable hop overlays. Search, overlay multicast, content delivery, and security are summarized in subsequent sections.

2 Overlay Basics

2.1 Classification and Taxonomy

The many different designs for P2P networks have led to various proposals for classification. For example, file sharing systems have been divided in to generations. First generation are hybrid designs that combine servers with P2P routing, and second generation are decentralized architectures. Anonymized P2P systems such as Freenet and I2P are sometimes referred to as third generation. Categorization by generations has several shortcomings. It leaves out many other important dimensions and doesn't explain what subsequent generations are likely to provide. Further, systems of all three generations were in use at the same time.

Another common distinction is to divide P2P overlays into unstructured and structured types. Unstructured overlays are usually further distinguished by how search requests are propagated, distribution of node degree in the peer population, and by differences in link formation with neighbor peers. Structured overlays are differentiated according to a variety of dimensions such as:

- maximum number of hops for routing a request (e.g., multi-hop, one-hop, variable hop)
- routing algorithm (e.g., prefix, XOR, geometric distance, address space difference, semantic distance)
- node degree with size of overlay (e.g., constant degree, logarithmic degree)
- overlay geometry
- lookup type (iterative vs recursive, and serial vs parallel)

Beyond the unstructured and structured categories, we find other categories, such as hierarchical overlays, federated overlays, overlays for deploying network services called service overlays, overlays for sensor grids called sensor overlays, overlays which route queries by semantic relationships called semantic overlays, and overlays providing support for mobile nodes in IP and ad hoc networks. We describe

these categories in more detail in the following sections. Figure 2 shows a classification tree for many of these categories of P2P overlays. Classifications for mobile-enabled overlays, services overlays, and sensor overlays will be discussed in later sections.

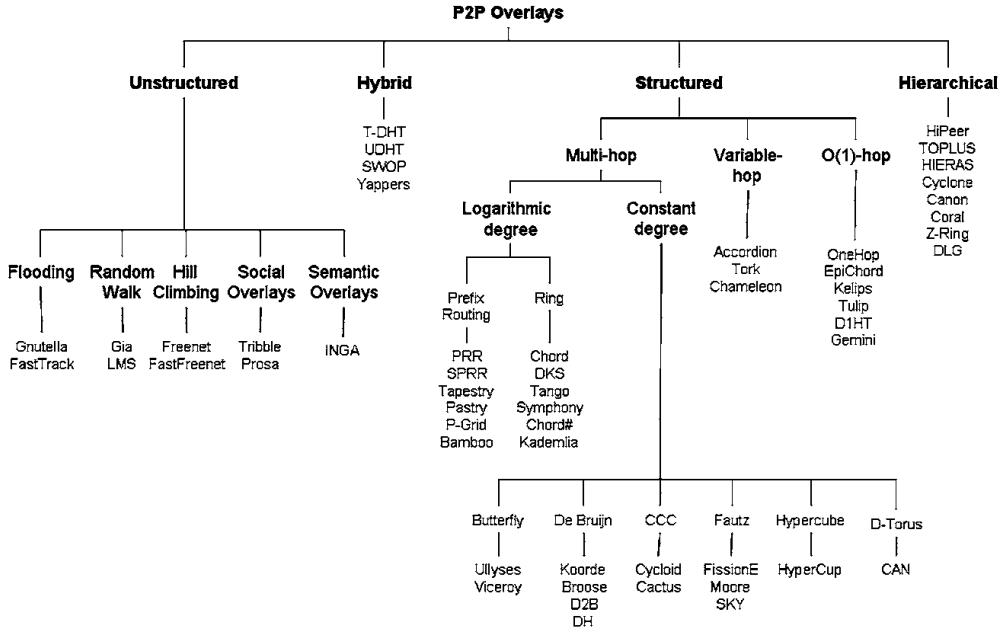


Fig. 2 Classification of P2P overlays

2.2 Unstructured Overlays

An unstructured overlay is "an overlay in which a node relies only on its adjacent nodes for delivery of messages to other nodes in the overlay. Example message propagation strategies are flooding and random walk" [26]. The graph structure formed by unstructured overlays can be compared to that of random graphs, scale-free or power law random graphs, graphs exhibiting small world phenomena, and other social networks. An important research focus for unstructured overlays has been the design of efficient search, including query propagation, object placement, and query processing. More details about search are discussed in a later section.

Another important research focus has been the optimal graph structure for unstructured overlays, and decentralized algorithms to form and maintain these graphs structures under changing peer and object populations. Influential unstructured overlay designs are listed in Table 3.

Table 3 Example unstructured overlays

Type	Design	Features	References
Hill climbing backtracking	Freenet	Routing using hill climbing with backing, and providing security, anonymity, and deniability	[27, 28]
Hill climbing backtracking	Fast freenet	Extension to Freenet in which objects stored at each peer are summarized and summaries are shared with neighbors	[29]
Hill climbing backtracking	Small world freenet	Freenet augmented with links emulating small world model	[30]
Flooding	Gnutella	Superpeers use flooding of requests on behalf of regular peers	[31, 32]
Random Walk	Gia	Uses techniques such as dynamic topology adaptation, active flow control, one-hop index replication, and biased random walk to improve performance	[33]
Flooding	FastTrack	Proprietary protocol with superpeer architecture that uses connection shuffling	[34]
Random Walk	LMS	Local minima search proactively replicates objects using consistent hashing of object identifiers to place objects at close node identifiers	[35]
Hybrid	SWOP	Structured overlay with additional cluster and long links to emulate small-world properties	[36]
Semantic routing	INGA	Semantic overlay in which each peer organizes a semantic index for its content, and queries are routed according to the associated topic, and evaluated using a semantic matching function	[37]
Preference directed queries	Tribler	Social-based overlay in which peers exchange preference lists to exploit social affinity between peers with similar preferences	[38]
Relevance directed queries	PROSA	P2P resource organization by social acquaintances manages peer links according to semantic strength of the relationship of the respective peer interests	[39]
Hybrid	Yappers	Combines local nodes in to small DHTs, and routes between DHTs using unstructured links	[40]

2.3 Structured Overlays

A structured overlay is: "an overlay in which nodes cooperatively maintain routing information about how to reach all nodes in the overlay" [26]. Compared to unstructured overlays, structured overlays provide a limit on the number of messages needed to find any object in the overlay. This is particularly important when searching for infrequently occurring or low popularity objects. In order to provide deterministic routing, peers are placed into a virtualized address space, the overlay is organized into a specific geometry, and a converging distance function over the combined object and node identifier space is defined for the routing forwarding algorithm.

Each peer has a local routing table which is used by the forwarding algorithm. The peer's routing table is initialized when the peer joins the overlay, using a specified bootstrap procedure. Peers periodically exchange routing table changes as part of overlay maintenance. Overlay maintenance is discussed in a later section.

The majority of structured overlays use key-based routing in which "a set of keys is associated with addresses in the address space such that the nearest peer to an address stores the values for the associated keys, and the routing algorithm treats keys as addresses" [26]. A distributed hash table (DHT) is a structured overlay that uses key-based routing for put and get index operations and in which each peer is assigned to maintain a portion of the DHT index.

Because the address space is virtualized and peer addresses are typically randomly assigned, peers which are neighbors in the overlay can be distant in the underlying network. While this improves the fault tolerance of the overlay, it causes significant performance loss. Consequently, *topology-aware overlays* use measurements of proximity of peers in the underlying network to create neighbor peers in the overlay.

There has been some interest in the efficient support of broadcast in structured overlays. Broadcast can be used for group communication, blind search, and overlay configuration. Example approaches are defined in [41–44].

Table 4 summarizes many of the designs for structured overlays.

Table 4 Structured overlays by category

Type	Design	Features	References
Prefix routing	PRR	First DOLR algorithm, used suffix based routing	[45]
	SPRR	Added join/leave and maintenance to PRR	[46]
	Tapestry	Based on PRR with an added join/leave and maintenance mechanism	[47]
	Pastry	Prefix routing with last hop using neighbor table	[48]
	P-Grid	Prefix routing	[49]
	Bamboo	Variation of Pastry used in OpenDHT	[50]
	Z-Ring	Hierarchical address space with large base to reduce latency	[51]
Logarithmic degree	Chord	Logarithmic spaced links to neighbors around predecessor-successor ring, consistent hashing, unidirectional requests	[52]
	DKS(n,k,f)	Distributed k-ary search with routing region at each hop divided into k regions. $k = 2$ similar to Chord	[53]
	Tango	Variation of DKS which reduces links to increasing scalability	[54]
	Chord #	Modification of Chord which replaces consistent hashing with key-order preserving indexing	[55]
	Symphony	Bi-directional routing with added long-links to shorten lookup hop count	[56]
	Kademlia	XOR distance function, parallel requests	[57]
Fixed degree	CAN	D-torus with cartesian coordinate system	[58]
	Viceroy	Butterfly	[59]
	Ulysses	Butterfly	[60]
	Cycloid	Cube connected cycle, prefix-style routing	[61]
	Cactus	2 trees combined with cube connected cycle	[62]
	Koorde	de Bruijn, based on Chord	[63]
	Broose	de Bruijn	[64]
	D2B	de Bruijn	[65]
	DH	de Bruijn	[66]
	Hi-Peer	Multi-ring de Bruijn	[67]
	Hypercup	Hypercube	[68]
	FissionE	Kautz graph	[69]
	Moore	Kautz graph	[70]
	SKY	Kautz graph	[71]
	DLG	A universal framework for building DHTs based on arbitrary constant-degree graphs, using distributed line graphs	[72]

(continued on following page)

Table 4 (continued) Structured overlays by category

Type	Design	Features	References
O(1)-hop	EpiChord	Iterative parallel lookup with opportunistic maintenance	[73]
	OneHop	Peers organized into slices and units, requests routed through slice and unit leaders, with active maintenance	[74]
	Kelips	Epidemic multicast protocol for overlay maintenance	[75]
	Tulip	2-hop overlay similar to Kelips which adds locality awareness	[76]
	Gemini	2-hop with high probability, combines suffix and prefix routing	[77]
	D1HT	Uses active maintenance algorithm EDRA where all join/leave events are forwarded to all other peers in logarithmic time	[78]
Variable Hop	Accordion	Recursive parallel lookup with bandwidth adaptive maintenance	[79]
	Chameleon	Hybrid of EpiChord and D1HT	[80]
	Tork	Hybrid of EpiChord and D1HT	[81]
Hierarchical	Multiple rings	Multiple overlays connect to a super-ring overlay, and use hierarchical routing to route requests between overlays	[82]
	TOPLUS	Peers organized into groups, each with own overlay; higher-level overlay is defined among the groups; intra-group routing used to route between overlays	[83]
	HIERAS	Overlay divided into several rings, with peers in low level rings selected according to locality	[84]
	Cyclone	Leaf overlays are connected in one horizontal overlay	[85]
	Z-Ring	Prefix routing with base = 4096, overlay organized in to groups for reduced maintenance	[86]
	Canon	At each successive level an overlay is formed which contains all peers in the subsumed overlays in the lower levels	[87]
	Coral	Peers are members of successive clusters of overlays, and lookups use distributed sloppy hash tables (DSHTs)	[88]

2.4 Hierarchical and Federated Overlays

A *hierarchical overlay* is an overlay architecture that uses multiple overlays arranged in a nested fashion, and the nested overlays are interconnected in a tree. A message to a peer in a different overlay is forwarded to the nearest common parent overlay in the hierarchy. When large scale distributed systems exhibit locality in their operation, hierarchical structure can increase overall performance. Hierarchical overlays exhibit hierarchical organization in addressing and routing. Different overlay regions in the hierarchy may use different routing algorithms. Important requirements for efficient operation of hierarchical overlays include avoiding bottlenecks and keeping the hierarchical structure balanced. Examples of hierarchical overlays include [82–88].

A *federated overlay* [89] is an overlay that is formed from a collection of independent overlays, each implemented by a separate administrative domain, and which may use different routing algorithms and addressing mechanisms in each domain. Each overlay is autonomous, and messaging operations between overlays require peering arrangements. Each domain manages the authentication, authorization and other management tasks for its overlay. Federation offers one mechanism by which overlay operators can offer specialized services to their customers while still providing the benefits of scale. An important requirement for federation is the trust relationship between each domain. Similar to the peering relationships between service providers for the Internet backbone, security breaches in one network have the potential to cascade to other overlays through the peering points. Thus the least secure overlay in the federation becomes a vulnerability for the remaining overlays.

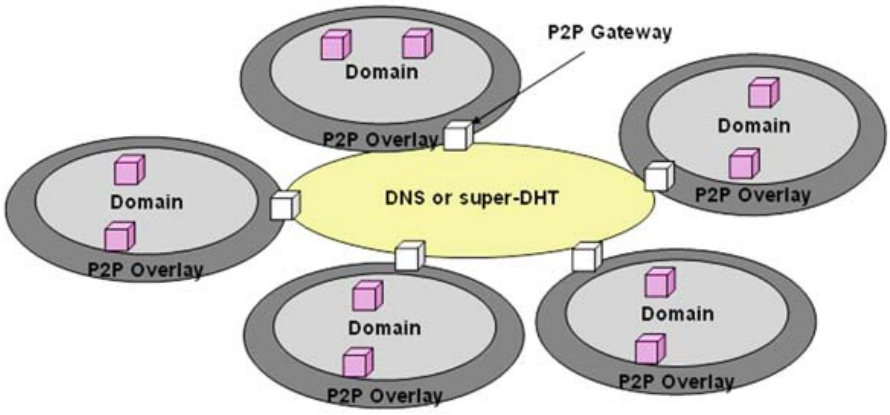


Fig. 3 Example of a federated overlay [89]

Individual overlays operate as usual according to the specific overlay algorithm. Pairs of overlays connect through gateway peers (Fig. 3). Gateway peers can discover other gateway peers for other overlays by different means such as a lookup in an interconnecting overlay or by DNS. A peer sending a message to a remote overlay first discovers a gateway peer and sends the message to it for forwarding. A multipart address scheme is used to distinguish objects and peers in separate

overlays. Figure 4 shows an example message forwarding in a federated overlay using different types of overlays. Let's compare the routing in a federated overlay, each of size n peers, with that where all peers belong to a single overlay of size $N = \sum n$.

Case 1: A look-up within the same overlay will perform better than in the global overlay due to the smaller number of hops.

Case 2: A look-up crossing multiple federated overlays will perform worse as the number of overlays increases due to the overhead of address resolution and routing at each peering point. Increasing the number of direct peering relationships improves performance at the cost of additional complexity in forming and managing each peering point.

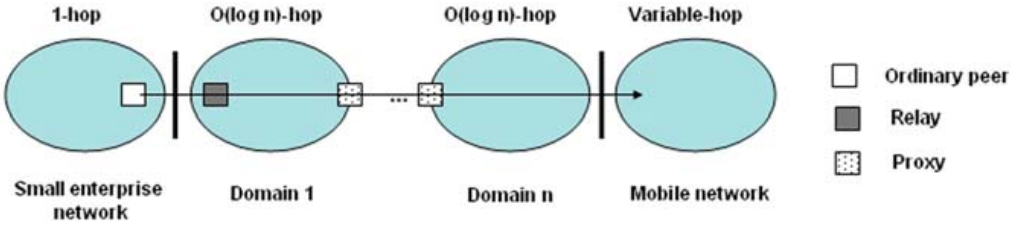


Fig. 4 Example multi-overlay messaging in a federated overlay

2.5 Service Overlays

As the Internet has grown, the need for new network services has also increased. However there is usually a long delay in developing and deploying any new service or extension to an existing service if it requires changes to network layer protocols, routers or other network infrastructure. This is due to the need to insure interoperability and avoid the introduction of new security vulnerabilities. To accelerate the deployment of new services and avoid changing the network infrastructure, many network services have been implemented as application layer protocols using end systems attached to the network. This includes multicasting, VoIP, and content delivery networks (CDNs). When an overlay is used as the basis for such application layer services, it is referred to as a *service overlay*.

In addition, service orientation is a new paradigm for architecting web applications and distributed enterprise systems. One of the shortcomings of P2P applications to date is that each application has had a dedicated overlay designed specifically for it. The idea of applying service orientation to P2P overlays is also referred as a service overlay. From this perspective, P2P applications are modularized as services which operate at a layer above the overlay, each service interface is defined using a service description, and the overlay provides a generic service discovery and advertisement mechanism. As the number of P2P applications increases, this has the advantage of enabling the reuse of the same P2P infrastructure for a collection of

Service Overlays

Network Service	Overlay Example
Routing	Resilient Overlay Network Routing Overlay
Domain Name Service (DNS)	DNS via DHT
Multicast	Application Layer Multicast
Content Delivery	Stream Based Overlay Network (SBON)
AAA	Peering of RADIUS Domains
QoS	QoS Aware Overlay
Session Establishment	IETF P2P-SIP
Telephony Services	Relays Feature Servers

Service Oriented Architecture	Overlay Example
Service Discovery & Advertisement	INS/Twine
Service Composition	SpiderNet
Middleware	NEMO
Service Models	
Load Balancing	Beehive

Fig. 5 Types of service overlays [26]

applications. In an open P2P platform, it could also lead to the delivery of many new 3rd party applications.

Figure 5 shows examples of these two categories of service overlays. Further information on service overlays for routing, called resilient overlays, can be found in [90–94], and research on using overlays for DNS is discussed in [95–98].

2.6 Semantic Overlays

A semantic overlay is an overlay network in which routing topology is organized according to the semantic associations and relationship of information being stored in the overlay. Similar to the semantic web, content can be stored and accessed using a semantic model that is more convenient for the user. Several semantic overlays have been proposed such as Inga [37]. Challenges facing semantic overlays include:

- Agreeing on common ontologies within a community of peers
- Updating the ontology across the distributed set of peers when new concepts and relationships become important.
- Efficient semantic matching for object placement and search
- Implementing semantic behavior directly in the overlay routing mechanism versus layering it on top of an existing DHT or unstructured overlay query mechanism

2.7 Sensor Overlays

A sensor overlay is an overlay network that connects elements of a sensor infrastructure or grid. The purpose of a sensor overlay is to hide physical layer network constraints from applications, and to make data collection and infrastructure control logically separate from the physical layer routing. In addition, in a large sensor grid, there may be multiple planes, each with different physical routing layers and data collection. The sensor overlay can potentially unify these and make integration with conventional overlays simpler. An example sensor overlay is PIAX [99].

2.8 Research Directions

Many variations of overlays have been studied to date. Efforts continue to reduce latency, better adapt the overlay routing to the semantics of the applications, improve load balancing and response to flash crowds and similar dynamics, adapt to changing network conditions, and increase the ability of the overlay to self-organize.

3 Overlay Dynamics, Heterogeneity and Mobility

3.1 Churn and Overlay Maintenance

Peers may join or leave the overlay at any time. Overlays use join and leave protocols so that neighbors can update their routing state, and so that newly joined peers can quickly make connections with active neighbors. A candidate peer needs to discover an existing peer by which to join the overlay. The process of discovering and contacting an existing peer is called *peer bootstrap*, and involves mechanisms outside of the overlay such as contacting a well-known bootstrap server or making local broadcast announcements. When a peer joins the overlay, it typically receives its initial routing and object state from one or more peers designated by the bootstrap peer. After that, the peer modifies its state based on the operation of the overlay protocols. When a peer leaves the overlay it may signal its neighbors using a leave protocol. The neighbors then make changes to their routing state, and object state may be migrated or replicated as well. If a peer is disconnected without notification, neighbors use a heartbeat mechanism to detect the departure and trigger the corresponding routing and object state updates.

Churn is the arrival and departure of peers to and from the overlay, which changes the peer population of the overlay. *Overlay maintenance* is the operation of the overlay to repair and stabilize the overlay routing state in response to churn. The overhead for overlay maintenance increases as the churn rate increases. It also increases proportional to the routing state maintained by each peer, which is in turn proportional to the size of the overlay and the degree of each peer. There are techniques to

reduce churn itself, such as incentives for peers to stay connected to the overlay. In addition, newly joined nodes can be quarantined, treated as client-only nodes, either due to limited capacity or until the peer reaches a lifetime threshold. This relies on the peer lifetime distribution being heavy tailed, which has been found to occur in practice.

Empirical data on churn in operational P2P applications has been gathered by deploying overlay crawlers. Crawlers connect to many other peers in the overlay in order to gather a snapshot of overlay membership. By continuing the process, information about membership changes and peer connectivity can be gathered. For P2P file sharing systems, measurements show a median peer lifetime of less than 1 hour and as little as 3 minutes, while as many as 2% of the peers will have a lifetime as long as one day [100]. On the other hand, peer lifetime measurements for Skype superpeers show a much longer median lifetime of about 5.5 hours [100].

Overlay maintenance can be classified as active or opportunistic. In active maintenance, routing table maintenance operations are triggered on peer join and leave events. An example active maintenance algorithm is EDRA (Event Detection and Recording Algorithm) used in D1HT [78]. As the churn rate changes, the routing state updates change proportionally. In opportunistic maintenance, routing table maintenance is performed as part of peer request routing or if the routing state falls below a minimum threshold. If peer request rates are high, for example for object lookups or inserts, then the routing state will be updated more frequently. An example opportunistic maintenance algorithm is that used in EpiChord [73].

While most analysis of overlays assume steady state is reached in which overlay maintenance matches the churn rate and the routing state enables the desired overlay geometry, there is growing thought that such an overlay state is not reached in practice, particularly for large overlays, due to the continuous changes to peer membership and time needed to propagate membership changes throughout the overlay. More likely, peers are not only out of sync with respect to the actual membership of the overlay but also have inconsistent routing state with other peers. To more accurately reflect the dynamics of the overlay, stochastic models of overlay membership have been developed for specific overlays, for example, [102, 103].

3.2 Mobility in P2P Overlays

While mobile nodes represent a small percentage of overlay peers today, in the future as the capability and network bandwidth increases and the population of such devices grows, this situation may be reversed. Thus the impact of mobility on the performance of the overlay is an important question [119].

Mobile devices have four properties that affect their interaction with the overlay in ways different from conventional desktop computers: roaming, energy limitations, node heterogeneity, and multi-homed interfaces. Network roaming causes IP address changes, and in conventional overlays, re-binding the overlay address to the IP address is effectively a leave-join sequence, leading to mobility-induced churn.

Approaches to mitigating mobility-induced churn include use of Mobile IP at the native layer, treating mobile nodes as stealth nodes [104], and designating a non-mobile node as virtual home agents for a mobile node [105]. Energy limitations of nodes increase the likelihood of a node going into a stand-by state. In today's overlays, this is likely to cause a node disconnect from the overlay.

Node heterogeneity means that nodes will not have equal capacity to store objects, participate in overlay maintenance, relay traffic from other nodes, and so forth. Variable hop overlays are one way to address these variations, and are discussed later in this section.

Multi-homed nodes are nodes that can connect to two or more different network interfaces at the same time. This could be used to provide redundant paths for peers to send and receive messages, which might reduce the impact of mobility induced churn.

3.3 Overlays for MANETs and Ad Hoc Networks

A mobile ad hoc network (MANET) is a set of mobile nodes which act as both routers and hosts in an ad hoc wireless network. The nodes route messages to other nodes without using a network infrastructure. Because of their limited power and capacity, MANET nodes transmit in range-limited broadcast messages which reach only nearby nodes. The MANET topology may change rapidly and in unpredictable ways.

As discussed in the introduction section of this chapter, integration of sensor grids, personal area networks, vehicular networks, and other ad hoc networks with Internet-based overlays is an important requirement for future global overlay based applications. Due to similarities between MANETs and the P2P model at both the application and network layer, there has been significant interest in adapting P2P overlays to work efficiency with MANET routing protocols. For example, many research systems have integrated flooding style unstructured overlays with MANETs. A summary of research activities is given in Table 5.

3.4 Heterogeneity and Variable Hop Overlays

A *variable-hop structured overlay* is a structured overlay that adapts the hop-count performance of the overlay according to the peer's network bandwidth budget so that at higher bandwidth budget the average hop count decreases and at lower bandwidth budget the average hop count increases. The performance of structured overlays depends on the accuracy and completeness of peers' routing tables, but more accurate and larger routing tables require more maintenance traffic. In addition, maintenance traffic grows with the churn rate and with the size of the overlay. Due to differences

in nodes' network and computational capacity, different approaches have been proposed to avoid having all nodes operate at the least common denominator level. The superpeer architecture which elevates the more capable and more reliable nodes to full status is a common approach. Variable hop overlays as demonstrated in [115] are another important direction.

Variable hop overlays take advantage of the ability of the overlay protocol to adapt its bandwidth utilization through changing configuration parameters. Each peer adjusts its routing table size and accuracy according to the available bandwidth at that peer. During periods where the nodes have low bandwidth capabilities, overlay routing performance may reach that of multi-hop overlays while for higher bandwidth, routing performance reaches one-hop.

In addition to Accordion [115], other proposals for variable hop overlays include Tork [116] and Chameleon [117, 118].

Table 5 Features of P2P overlays for MANETs [26]

System	MANET routing algorithm	P2P overlay	Lookups	Evaluation size (nodes)	Node speed (m/s) and range
MHT [106]	GPSR	None	Key maps to node's path	1000 to 100,000	10–15 m/s 2000×2000 m^2
Ekta [107]	DSR	Pastry	Prefix key-based	50	1–19 m/s 1500×300 m^2
MPP [108]	Extended DSR	Gnutella	Flooding	50 . . . 200	0–5 m/s $\leq 2000 \times 2000$ m^2
XL-Gnutella [109]	OLSR	Gnutella	Flooding with superpeers	50	≤ 15 m/s not stated
MADPastry [110]	AODV	Pastry	Prefix key-based with clustering around landmarks	100 and 250	1.4 m/s 1000×1000 m^2
FastTrack over AODV [111]	AODV	FastTrack	Flooding with superpeers	50	0–20 m/s 1500×320 m^2
ORION [112]	Neutral, AODV and SMB	Unstructured	Flooding	40	0–2 m/s 1000×1000 m^2
ISPRP [113]	DSR	Chord	Key-based	1000	NA
Dynamic P2P source routing [114]	DSR	Pastry	DP2PSR	800	9–19 m/s NA

3.5 Research Directions

Understanding the dynamics of large overlays in the face of changing peer populations, peer heterogeneity, peer mobility involves many challenging problems. While designing for heterogeneous peer populations involves relatively static peer distinctions, the ability to efficiently adapt peers to changing capacity and network conditions will involve considerations at a much smaller time scale. In addition, it remains an open question as to the practicality of large overlays if the majority of the devices are mobile. Finally, different models of node heterogeneity in terms of distribution and density in the overlay may become important in the design of overlay adaptation mechanisms.

4 P2P Content Access and Delivery

In recent years, the number of digital content, such as music and video, available on the Internet, the number of users accessing digital content through the Internet, and the number of digital video being streamed over the Internet each day are all growing exponentially. This obviously is placing an intense demand on the network bandwidth at the Internet backbone as well as on the servers that are offering the digital video and audio services. To improve content accessing scalability, Content Delivery Network (CDN) was invented and widely deployed in the last ten years. The evolution of CDN where a single sited content server is replaced by a set of distributed content servers placed strategically to provide not only better distribution of files but also better streaming of real-time media has many advantages. First of all, network congestion can be significantly reduced since servers are geographically distributed to better serve clients in given regions with low latency. Second, with servers placed at the edge of the network and closer to users, better quality of experiences can be expected for real time media streaming. The advantages of pushing content closer to end users suggest a natural extension of conventional CDN, P2P content delivery where contents are moved all the way to end users. It can potentially offer more advantages over traditional CDN, although it must be balanced with security, resource contention, and DRM issues.

Multicast is an effective content delivery method with reduced network bandwidth requirement. Due to cost issues, IP multicast has not been widely deployed. The explosion of streaming media applications thus offers another fertile ground for P2P based content delivery to blossom.

Taking advantage of the high scalability and the low cost in implementation properties of P2P networks, today P2P content access and delivery has become one of the most popular P2P applications. This includes P2P music sharing, P2P video sharing, P2PTV, P2P radio, P2P video streaming, etc. Just like in centralized content delivery systems, content can be delivered via downloading or steaming to the end users in P2P networks. At delivery, a media stream is segmented into data blocks that

are delivered via flooding, random walk, or via a topology defined specific route in the P2P overlay network. Depending on the network topology, content blocks may be forwarded along a distribution tree rooted at the source peer or flown through a mesh network. And differentiated by the initiator of the content delivery, the content blocks may be pulled or pushed from the source peer to the destination peer.

Although many P2P content delivery applications are seen today, many fundamental technical issues are still not fully resolved, for instance in content search, content streaming, and content caching and replication.

4.1 Content Search

Searching is a key step in data access. It is certainly the case in P2P content access as well. P2P networks take advantage of the distributed resources at peer nodes. Contents are scattered and duplicated in the P2P network in a distributed fashion. Hence, content retrieval in a P2P network needs to contemplate the specific network model as well as the characteristics of the content being accessed. Ideally, a P2P content search algorithm should comprise support of complex queries, low cost in implementation, and fast and high accuracy query return capabilities. Today, most structured P2P networks support static key and ID based object lookups while unstructured P2P networks can handle certain complex type of queries, such as range queries. Although semantic query and content based query can enrich user experiences in content search, they are hardly supported by any P2P content delivery systems today. This is because those types of queries are still posing significant technical challenges.

Content search schemes and capabilities are largely dependent on the content indexing and management schemes as well as the P2P network topology. Table 6 summarizes P2P indexing schemes. In a centralized indexing system where the index is kept at a centralized location in the P2P system, content searching is generally done by forwarding the query message to the centralized indexing server to facilitate object lookup. The server returns the lookup result which contains the location of the desired content object. Content is transmitted then in a P2P fashion. Localized, distributed, and hybrid indexing schemes can effectively reduce the risk of network disruption owing to their distributed nature. Care must be taken when designing the query scheme to reduce the cost associated with query message forwarding and flooding.

Many people associate DHT with P2P search. This is because DHT based object lookup is a widely adopted search scheme in structured P2P networks. Most DHT based schemes rely on numerical keys to index and query objects in the P2P network. Object searching is accomplished using key distance and routing towards the peer that has the closest key to the querying object key. It offers efficient key or ID based exact match lookups with guaranteed query returns. However, managing a consistent DHT requires considerable effort due to the dynamics of the network topology. Another drawback of DHT-based system is its inability to support

complex queries. In most applications, obviously, keyword, range, and semantic queries are more useful than key or ID-based exact match search.

Table 6 P2P indexing schemes

Indexing schemes	Index location	Query propagation	Content object delivery	Key limitations
Centralized indexing	Central server	Query is sent to the central server directly and resolved at central server	Querying peer obtains the content object location, i.e. the address of the source peer who has the content object; it then sends a request for delivery to the source peer directly; content object can now be delivered from the source peer to the destination peer directly	Vulnerability to attacks on the server and the possibility of bottleneck effect at the server
Localized indexing	Local peers	Query is propagated from peer to peer until the desired content object index is found	Content object may be delivered directly from the source peer to the query peer upon localization of the object	High cost associated with query flooding and low object retrieval efficiency
Distributed indexing	Distributed among peers	Query is forwarded to neighborhood peers based on peer routing table until the target object index is found	Querying peer obtains the location of the content object, sends a request to the source peer, and then receives the content object from the source peer	Possible delay at peer joining due to index set up
Hybrid indexing	Super nodes; super nodes and local peer nodes	Query is first searched locally at the local peer and the super node that is connected to the local peer directly. If content object is not found in local indices, query is propagated to other super nodes according to the routing table until it is found or a predefined Time-To-Live (TTL) threshold is reached	Querying peer obtains the location of the content object, sends a request to the source peer, and then receives the content object from the source peer	Powerful super nodes to sustain frequent query flooding are needed

Keyword search may be realized in P2P networks using the popular vector space model and/or inverted indexing based approaches. High cost associated with query flooding is one of the key drawbacks of those approaches if a non-centralized indexing scheme is employed. When the indices are distributed over peers, a simple query may cause a large amount of data being transmitted over the network. To reduce cost, one might take advantage of conventional information retrieval schemes. For instance, content summary based inverted indexing was proposed in [120]. Since a query can be processed by transmitting a much smaller candidate list, bandwidth demand for query flooding can be significantly reduced. The trade off is the additional storage space requirement and most importantly the reduction in recall rate in DHT based structured overlay. Today, how to implement efficient keyword search remains a challenge problem in P2P.

Compared to structured overlay, unstructured overlay has more freedom in implementing complex queries. Flooding, iterative depending, and random walk are commonly adopted searching approaches in unstructured P2P. Table 7 list the properties of these types of searching schemes in unstructured P2P networks.

Table 7 A comparison of searching schemes in unstructured P2P

Search scheme	**Characteristics**	**Cost**
Flooding	Query requests are flooded through the P2P network with the querying peer being the center of the flood	High. Massive amount of query messages are being transmitted for a single query
Iterative deepening	A growing ring is used to iteratively deepen the query flooding range until the target object is found [121]	High but lower than flooding based approach. Massive amount of query messages are being transmitted for a single query
Random walk	The querying node forwards (walk) the query message (walker) to one randomly selected neighbor which randomly selects its neighbor to forward the query message until the target object is located	Low to medium. In a K-random walk scheme, the cost is proportional to K while the delay is inverse proportional to K
Guided search	"Guidance" on where the query message should be forwarded is employed to improve query efficiency. Keyword vector, query similarity function, peer ranking and profile [122] may be used to guide the query forwarding	Low. Though recall rate can be significantly reduced if the "guidance" function is flawed

A query that retrieves all objects between an upper and a lower bound, i.e., an exclusive range, is called a range query. Likewise, a query that retrieves all objects within a multi-dimensional range is called a multi-dimensional range query. Methods to resolve range queries in classical database problems can be easily imported

into unstructured P2P networks. However, it is relatively harder to achieve in structured P2P due to the "clear-cut" nature of a DHT. If range attuned numerical keys can be generated, range query could be supported in DHT based system. Locality Sensitive Hashing (LSH) [123] is one such approach to hash similar data partitions to nearby identifiers and similar ranges to the same peer with high probability. Noticeably, LSH has poor scalability. SkipIndex [124], another partition based scheme that offers a solution to range query also demonstrated impressive results in small scale P2P networks. How to design a scalable range query scheme and how to design a scheme that can offer scalable and efficient multi-dimensional range query support for DHT based P2P remain challenges.

In a traditional information system, the most challenging types of searches are semantic search and content based search. This is certainly true in P2P networks as well. How to support efficient semantic search and content based search for multimedia content access will need considerable effort and continuous investigation.

4.2 P2P Streaming and Multicasting

Content delivery, based on the way the content is transported and consumed, can be categorized into downloading and streaming modes. Streaming refers to the delivery method where content is being consumed while it is being transported. Compared to download based delivery, streaming poses significant challenge due to the time bounded requirement.

One-to-one, one-to-many, and many-to-many are possible configurations in different P2P streaming applications. For instance, a live remote personal video sharing could be one-to-one or one-to-many, an Internet video application often takes advantage of a one-to-many configuration, and a video conferencing application is likely to involve many-to-many communications. Consequently, unicast, broadcast, or multicast protocols may be employed in different streaming applications.

In a P2P network, content can be streamed via a tree based or a mesh based overlay. In tree based approach, content, rooted at the source node, is pushed along the tree to the destination peers. Mesh-based overlays implement mesh distribution graphs for content streaming. In the mesh distribution graph, each new node first obtains a content block availability map where a set of randomly selected peers who have the desired content blocks are listed, it then contacts a subset of those 'good' peers to request for streaming, and obtains the content blocks from those peers based on a predefined protocol. PPLive and Coolstreaming, for example, both take advantage of mesh based streaming. While the mesh based pull model offers better load balancing capability, it often introduces additional delays due to the exchange of buffer maps.

Tree based schemes on the other hand entail considerable control overhead at peer churns. If any interior member leaves the group (the tree,) the tree is broken and the children of the failure or departure node need to be reconnected to the tree. These entail additional group management cost. Tree based system is also inherently

unbalanced. It does not utilize the bandwidth of the leaf nodes, causing a burden of duplicating and forwarding multicast traffic carried by a small subset of peers that are interior nodes of the multicast tree. This violates the fairness in resource and load sharing requirement in a P2P system. To improve fairness in resource sharing, multiple trees may be built to deliver different sub-streams. Splitstream [125], for instance, is one such scheme. Often this type of algorithms can work beautifully for a small scale P2P video streaming application. However, in a large scale P2P system, considerable complexity in building multiple balanced trees and tree reconnection at peer churns may significantly affect the system performance.

When there are multiple clients (receivers) simultaneously requesting/receiving the same media stream in a streaming application, multicast can be implemented. Multicast is a special type of streaming where protocols are defined to delivery a packet to a group of destinations at the same time using efficient strategies. Multicast can be deployed at different network layers. IP multicast which implements multicast at the IP routing level is generally high in implementation cost. P2P overlay multicast was invented to reduce deployment cost and improve scalability. A P2P overlay multicast system should implement [26]:

– Session identification
– Session initiation/creation
– Session subscription/join
– Session leave/graceful departure
– Session message dissemination/data forwarding
– Session fault tolerance/tree reformation at peer failure
– Session termination
– Session admission control
– Content access control and security

A comparison between P2P overlay multicast and IP multicast is given in Table 8. Obviously P2P networks offer considerable advantages, such as high scalability and low cost in implementation, for content streaming applications. The most considerable drawbacks in many commercially available P2P streaming and multicast video application systems such as PPLive include long startup delay and playback jittering. Quality of Experience (QoE), which indicates user experience and satisfaction, is a popular way today to measure the success of a content delivery service. Start-up delay and playback jittering are two important factors affecting user experiences. To reduce join and reconnection latencies in overlay multicast services, proximity based routing which improves arbitrarily long distances in routing hops is introduced [126]. Another approach [26] utilizes proactive step-parent selection to reduce the reconnection time in tree based multicast systems. That is, each peer locates its potential step-parent in advance. At tree reformation, a node that is a candidate parent immediately takes over the role of parenting. This cuts down real-time messaging needed for tree reconstruction, thus reducing the probability of playback jittering at the affected end hosts.

It was also shown that implicit protocols [127] where the control and data paths are defined simultaneously can support both latency-sensitive and high-bandwidth applications as well as very large group sizes.

Today, video streaming, one of the most popular means for content distribution, still suffer from several drawbacks when built on top of a fully distributed P2P overlay. These include high stress on the ISP links, dependency on high bandwidth peers, uneven quality distribution, lack of content security mechanisms and authentication capability, and long startup delay and channel switching delay. As a result, hybrid approaches are gaining considerable attention in the industry lately. Will hybrid solutions be able to offer reduced ISP stress and improved security as well as performance for large scale P2P streaming applications? Will hybrid systems ultimately solve the billing and accounting problem? These along with many other questions need to be resolved before P2P streaming takes on a full spin in commercial content delivery applications.

Table 8 Characteristics comparison: P2P overlay multicast and IP multicast

Metric	P2P overlay multicast	IP multicast
Efficiency	Relatively low	High
Stress on ISP	Relatively high	Low
Server bandwidth requirement	Significantly lower	High
Control overhead	Considerably higher	Low
Robustness	Generally lower	High
Lag between customers	Can be high	Low
Deployment cost	Usually very low	High

4.3 Caching and Replication

In P2P networks, data objects may be duplicated and saved temporarily or permanently on multiple peers. Caching and replication play key roles in reducing network bandwidth usage and origin server load and bandwidth requirement, reducing client side latency, and improving load balance, data availability, system reliability, and data access latency in a P2P network. Nevertheless, data consistency and synchronization issues need to be handled properly to reduce miss rate without high cost. The number of requests issued to peers for a particular content blocks and the frequency of cache replacement for instance, can affect the number of messages and network traffic pattern. Intuitively, flooding could guarantee object synchronization with the cost on additional communication messages and bandwidth requirement. Synchronization on demand could effectively reduce the communication cost,

however, it may only offer a weak guarantee. Can a joint flooding and on demand approach offer reasonable guarantee with acceptable overhead? This still awaits investigation.

Tables 9 and 10 compare several different replication schemes and caching schemes respectively.

Caching and replication in structured P2P could be tricky. Some of the structured P2P systems associate an object to the object identifier which is also the key to discover the location of the object. Effectivecaching and replication of

Table 9 P2P replication schemes

Replication scheme	**Characteristics**
Full replication	Data are replicated on all peers in a P2P network. All data are available to read locally. The cost to maintain the replicas is high
Partial replication	A portion of the content is replicated at some peers. Relatively lower maintenance cost with longer seek time
Synchronous replication	All replicas are simultaneously changed. Higher hit rate with high cost in replication synchronization is expected
Asynchronous replication	Delay in change at a remote replica is allowed. Cost in replication synchronization is reduced with reduced hit rate
Static replication	Replicas are fixed at all times
Dynamic replication	Location and number of replicas change by time, system condition, transactional status, etc.
Active replication	Locations of replica are predefined. Query request is sent to all replica servers
Passive replication	Peer nodes request and copy content from one and another. Passive replication processes query request sequentially and synchronizes the replicated copies of objects periodically
Random replication	Replicas are placed randomly at peers
Query path based replication	Content (data) object is replicated on all peer nodes on the path from the query destination back to the query source
Adaptive query path based replication	Object shall not be uniformly replicated on all peer nodes on the path. Instead, the object popularity, peer resources, etc. shall be taken into consideration to decide where the object will be replicated
Neighborhood replication	Data objects are replicated at some or all neighbor peers of a peer that holds the objects
Object location replication	The location of data objects are replicated in the neighbors of the peer that holds the data object whereas the data objects are not replicated

Table 10 P2P caching schemes

Caching scheme	Characteristics
Just-in-Time (JiT) caching	Immediately after a request is received from the client, the cache pulls the content from the server with the content sent to the cache and the requesting client simultaneously
Pre-Caching (PreC)	Contents are often cached before a request is received at the proxy

objects requires additional mechanisms. In Tapestry [47], replica roots identified with random keys which are generated using a replication function are used for object replication.

4.4 Summary of Design Issues

Cost, implementation complexity, efficiency, robustness, scalability, and quality of services and experiences are some of the key design criteria for P2P content delivery services. For instance, in a video conferencing service, the system has to meet the delay bound constraint to offer acceptable customer experience. Video applications are often resource demanding. Thus, system control overhead may have direct impact on a video application system's performance. Furthermore, system capability to cope with churn and network dynamics is imperative in any P2P content delivery systems. An efficient system that can take advantage of the P2P network resources in a fair and balanced way can have a strong impact on system scalability and performance.

4.5 Research Issues

Noticeably, many popular P2P related brand names are associated with content delivery. For instance, Kazaa offers music sharing, PPLive provides P2P based TV service, and Pando presents video downloading, streaming, and sharing capabilities to its customers. Although tens of similar P2P based content delivery services are available today, major content providers, network service providers, or telco companies have not been deploying P2P based content delivery systems. Why? From technology point of view, there are many technical issues still need to be resolved before P2P network takes on a full spin in content delivery. Security, efficiency in searching and delivery, fairness in resource sharing, and billing and accounting, for instance, are some popular issues.

5 Security

5.1 The P2P Security Concern

Security is not just an issue in P2P content delivery, but also an important issue in almost all types of P2P applications. In fact, information security has been a daunting subject area in today's networked world. Information security aims at safeguarding information and information systems through a range of policies, strategies, security products, technologies and procedures. It includes the protection of information and system's availability, confidentiality, privacy, and integrity. Today, relying on personal computer and the Internet for information storage, retrieval and asset management is becoming an everyday practice for many individuals; whereas for many organizations, the network is a primary and mission critical components whose day-to-day operation must be fully warranted. Hence network vulnerabilities have significant impact on enterprise as well as personal information security today. With the growing frequency and types of threats, from viruses to Trojan horses, from adware to spyware, from denial of service to distributed denial of service, from fraud to identity theft, ... people are more and more aware of the security threats and more and more concerned with various security threats that are presented to them via different networks and applications.

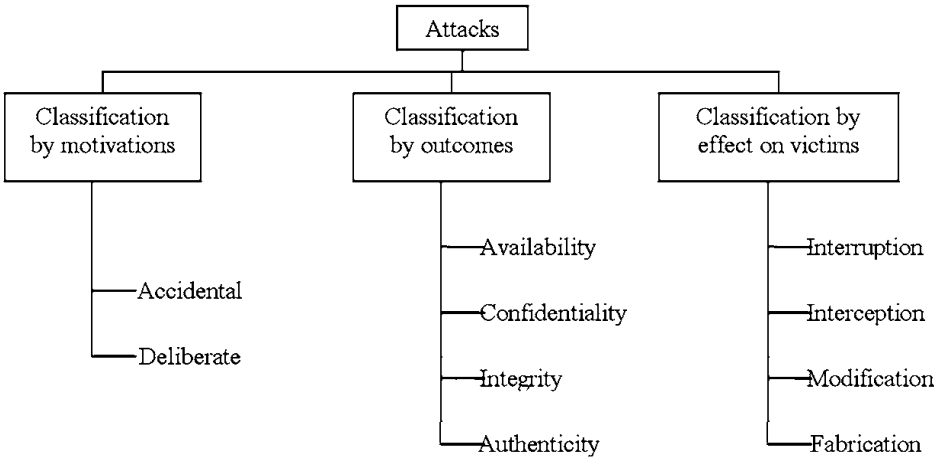


Fig. 6 Sample categories of attacks

With no central authority governing the authenticity and integrity of the sharing content and peers and with limited mechanisms for protecting the rights of content owners or the security of the client systems, P2P overlay network and applications add another dimension of security concern. Obviously, sharing files or computing resources on one's device with unknown peers over the Internet goes against many of the basic principles of securing your information. It could open up new doors for cyber criminals to steal confidential information, to damage personal or enterprise properties, and to poison the network for criminal intents.

5.2 Basic Classifications of P2P Network Security Threats

Through the years, many terms describing the security threats of networks or information systems were used. Some may be widely adopted and some may be confusing. Figure 6 and Table 11 list several popular categories and sample types of threats [26, 128–130].

Attacks on P2P network and systems may target at the overlay or the application layers [26]. Content theft, confidential information theft, service attacks, and computing and network resource theft and attacks are just some of the many types of threats P2P networks and application users are facing. Table 12 shows several unique P2P overlay security attacks defined in [131]. Those attacks could potentially

Table 11 Sample types of attacks

Category	Description	Sample attacks
Accidental	Security leakage caused by accidents	Accidental bandwidth clogging
Deliberate	Attacks are deliberate for criminal intent	Theft
Availability	Attacks that compromise the availability of the system or information, i.e. attacks that cause system or data unavailability to authorized or normal usage	Denial of service, bandwidth clogging, worm (e.g., using up the computer's resources and possibly shutting the system down)
Confidentiality	Attacks that compromise the confidentiality of the system or information, i.e. unauthorized disclosure of information	Theft, Trojan (e.g., one that logs keystrokes to steal information, RAT(Remote Access Trojan)) Spyware
Integrity	Attacks that compromise the integrity of the system or information, i.e. attacks that modify data without authorization	Virus
Authenticity	Attacks that compromise the authenticity of the system or information, i.e. attacks that target at or affect the genuineness of the information or the system	ID attacks, password attacks
Interruption	Unauthorized disruption	Distributed denial of service, bandwidth clogging
Interception	Unauthorized access	Theft, password attack, IP spoofing, nodeId attacks, Sybil attack
Modification	Unauthorized tampering	Reverse engineering
Fabrication	Unauthorized creation	Content/email spoofing

Table 12 Sample types of P2P overlay attacks [26, 131]

Category	Description
NodeId attack	Nodes obtain specific nodeId(s) (node identification) for malicious intent
Sybil attack	A small number of entities counterfeiting multiple peer identities so as to compromise a disproportionate share of the system
Message forwarding attack	Attacks that alter the route or the content of the message being forwarded for malicious intent
Routing table attack	Attacks that manipulate routing table entries for malicious intent
DDoS attack	Nodes work together to prevent a system from performing its task

jeopardize the availability, confidentiality, integrity, and/or authenticity of content, data, a system, a network, or a peer.

A P2P network is a particularly attractive platform for attackers to steal confidential information and to spread viruses. It often opens up a back door for hackers to easily gain access to devices and information that normally can not be accessed. For instance, a hacker can use a software tool, such as Wrapster, to disguise a confidential document. The confidential document, now appear to be a legitimate media content, such as an MP3 file, bypasses the enterprise security mechanisms and policies and is shared and transmitted through a P2P file sharing system. The receiver outside of the enterprise network can now unwrap the file and convert it into its original format. A piece of code, the virus, could also appear to be a popular file-sharing program and subsequently when downloaded, the virus gains access to the peers' data, information, and software on the device. Modifying data and files and destroying the file system are just two of the many damages a virus could cause. P2P networks also provided a fertile ground for attacks to cash in on a collection of peer resources to achieve malicious means. Distributed Denial of Service (DDoS) attack and Sybil attack are two most representative ones in this category. Denial of Service (DoS) attacks could cause service breakdown through disruption of physical network components; consumption of resources such as storage, computation, or bandwidth resources; obstruction of communications; and interference with configuration and state information. For example, a DoS attacker may use malware to max out a user's CPU time or crash a system by triggering errors in instructions.

5.3 Counter Measures

Leaking confidential information through P2P networks and applications is a primary concern at many organizations. In addition, bandwidth clogging, viruses, copyright infringement, etc. are also serious threats at the enterprise network level. Detecting and stopping P2P applications at the enterprise network level is a straightforward practice and is implemented by many organizations today. Notice

though if a laptop is used in a P2P application while disconnected from the enterprise network, it may still introduce security breaches when it is reconnected into the enterprise network. For instance, P2P software vendors or application service providers may offer free P2P file sharing or other services. In order to generate revenue, they may bundle advertiser applications or activity trackers to its P2P application and services. The advertisement or activity tracking application software maybe piggybacked in its application program, much like an adware or spyware could be. Obviously, other types of adware and spyware may also be piggybacked. They are often downloaded without users' knowledge and may run in the background even when the peer machine is disconnected from the P2P network. This opens up a window for hackers and introduces various potential risks. For instance, the adware or spyware may bypass the enterprise network firewall when the machine enters the enterprise network. It may contain viruses or worms that will spread around the enterprise network once the machine is reconnected. It may contain other security flaws and it certainly will use additional resources including computational, storage, and even bandwidth resources. To better address security concerns from P2P, one must be able to stop P2P activities on their networked systems and every component of the systems completely. To do this, strong policies should be implemented to allow automated protection on each and every component of the network. It includes protecting all components from becoming nodes in P2P networks while they are on and off the enterprise network.

While banning P2P maybe an easy solution to protect ones network from attacks caused by P2P networks and applications, it's certainly not a viable solution for all networks. To fight against P2P attacks while maintaining the privilege to employ some P2P applications, specific counter measures may be designed and implemented. A semi-decentralized P2P system, for instance, may help to reduce many types of P2P security risks. In a semi-decentralized P2P system, a centralized trust entity maybe utilized for security administration. Noticeably, the centralized authority can also become the victim of P2P attacks, such as a DDoS attack, if proper counter measure is not taken. Similarly, hybrid P2P can effectively reduce some P2P security risks while the super peers in the hybrid P2P network may become the victim of DDoS attacks if proper counter measure is not employed.

Obviously, fully distributed security mechanisms are needed in fully decentralized P2P systems. Castro [132] introduced secure routing table maintenance, secure nodeId assignment, and secure message forward as several primitives for secure message routing in structured overlay. By imposing strong constraints on routing table, binding nodeIds to node IP address, message authentication, and some other mechanisms, improved security at message routing in structured overlay is expected.

Studies on P2P DDoS attacks show that pattern detection and advanced filtering mechanism may be helpful in detecting DDoS attacks, the explosion of new types of DDoS attacks making it one of the toughest to defend [133, 134]. In [133], Mirkovic suggests to deploy comprehensive protocol, system security mechanisms and abundant resources to improve the system resilience to DDoS attacks.

5.4 Fairness, Trust and Privacy Issues

Adar [135] reported a 70% free rider testing result in Gnutella. Gnutella is certainly not alone in experiencing P2P free riding where peers are consuming resources without fair contribution of their resources. To improve fairness in P2P resource sharing, auditing, incentive, and micro-payment based mechanisms are proposed.

A P2P system relies heavily on a set of distributed peers working properly and fairly together. Today, a large scale P2P system can be thousands to millions in size with peers interacting with unknown peers. How can peers establish and maintain trust between one and another in a P2P system especially a large P2P system? How to perform peer authentication? A centralized trust management entity could solve the problem and yet it tends to be a single point of attack. A hybrid system although does not suggest a single point of attack, could still become impaired when attacks are targeted on several super peers at the same time. Distributed trust management, on the other hand, could impose high cost and overhead for pair-wise peer authentication.

Privacy in P2P networks is another constantly raised issue today. Most P2P networks do not implement appropriate privacy governance mechanisms. While anonymous communication offer means to protect privacy, it is offset by the risks in trust and reduction in communication efficiency.

5.5 More on P2P Security

Due to the autonomous and distributed nature and the wide availability of replicated objects, making a P2P network secure is a big challenge. Exposure to theft, distributed viruses, worms, Trojan horses, spyware, or DDoS attacks are just some of the many types of P2P attacks we are facing today. With decentralized security mechanisms not fully in place and traditional server-based security schemes not offering suitable means for fully decentralized P2P network protection, P2P security remains a daunting subject in the P2P research field.

As introduced earlier in this chapter, many P2P systems are designed for a specific application. A systematic counter measure with clearly defined security goals and security schemes that are application and system driven and carefully designed protocols and systems based on appropriate security policies, coupled with security educated P2P system and application users, perhaps can be expected to improve P2P system security and defend against various attacks. To further understand the security risks and counter measures in P2P networks, additional discussions and references can be found in Chapter 14 of [26].

6 Summary

There has been a great deal of research to devise and improve on a large range of overlay-related design dimensions. Also the number of applications of overlays is increasing due to growing capacity of wireless networks and end devices, and the popularity of P2P applications among end users. We have attempted here to summarize and highlight the key results to date. Nevertheless there are certain major questions about the P2P paradigm and its use that remain including:

- What is the significance of the P2P paradigm as a general distributed systems architecture, and how will it evolve in practice with respect to the client-server paradigm?
- What are the barriers to adoption of the many research results by deployed systems, and how can these be avoided?
- Is the first mover advantage in P2P applications surmountable, and can balkanization of the P2P landscape be avoided?
- What is the likely long-term impact on internet architecture and service providers?

Further sources of surveys on the field of P2P networks include [136–139] as well as the books [1, 140].

References

Introduction

1. D. Clark, B. Lehr, S. Bauer, P. Faratin, R. Sami, and J. Wroclawski. Overlay networks and the future of the Internet, Communications & Strategies 63, 3Q2006.
2. D. Clark, B. Lehr, S. Bauer, P. Faratin, R. Sami, and J. Wroclawski. The growth of Internet overlay networks: implications for architecture, industry structure and policy, 33rd Telecommunications Policy Research Conference, Sept. 2005, available at http://web.si.umich.edu/tprc/papers/2005/466/TPRC_Overlays_9_8_05.pdf.
3. L. Peterson, T. Anderson, S. Shenker, and J. Turner. Overcoming the Internet impasse through virtualization, IEEE Computer, Apr. 2005, 62-69.
4. The SpoVNet Consortium, SpoVNet: An architecture for supporting future Internet applications, www.spovnet.de.
5. Ambient Networks Deliverable, System design of SATO & ASI, www.ambientnetworks.org/Files/deliverables/D12-F.1_PU.pdf, Sept. 2007.
6. R. Devine. Design and implementation of DDH: a distributed dynamic hashing algorithm, Proceedings of the 4th international Conference on Foundations of Data Organization and Algorithms, Oct. 13-15, 1993, D. B. Lomet (ed.), Lecture Notes in Computer Science, Vol. 730, Springer-Verlag, 101–114.
7. W. Litwin, M.-A. Neimat, and D. A. Schneider. LH: linear hashing for distributed files, ACM SIGMOD Record, 22(2), June 1, 1993, 327–336.
8. W. Litwin, M.-A. Neimat, and D. A. Schneider. LH: a scalable, distributed data structure, ACM Transactions on Database Systems (TODS), 21(4), Dec. 1996, 480–525.
9. C. G. Plaxton, R. Rajaraman, and A. W. Richa. Accessing nearby copies of replicated objects in a distributed environment, Proceedings of the 9th annual ACM symposium on parallel algorithms and architectures, Newport, RI, June 23-25, 1997, 311–320.

10. D. Karger, E. Lehman, T. Leighton, R. Panigrahy, M. Levine, and D. Lewin. Consistent hashing and random trees: distributed caching protocols for relieving hot spots on the World Wide Web, Proceedings of the 29th Annual ACM Symposium on theory of Computing, El Paso, Texas, May 4-6, 1997, STOC '97, ACM Press, 654–663.
11. M. Ripeanu, I. Foster, and A. Iamnitchi. Mapping the Gnutella Network: Properties of Large-Scale Peer-to-Peer Systems and Implications for System Design, IEEE Internet Computing, 6(1), February 2002.
12. S. Saroiu, P. K. Gummadi, and S. D. Gribble. A Measurement Study of Peer-to-Peer File Sharing Systems. In Proceedings of Multimedia Computing and Networking 2002 (MMCN'02) (San Jose, CA, Jan. 2002).
13. S. Sen and J. Wang. Analyzing Peer-to-Peer Traffic Across Large Networks. In Proceedings of the ACM SIGCOMM Internet Measurement Workshop 2002 (Marseille, France, Nov. 2002).
14. Y. Chawathe, S. Ratnasamy, L. Breslau, S. Shenker, and N. Lanham. GIA: Making Gnutella-like P2P Systems. Scalable. ACM SIGCOMM 2003.
15. Y. Qiao and F. Bustamante. Structured and unstructured overlays under the microscope: a measurement-based view of two P2P systems that people use. In Proceedings of the Annual Technical Conference on Usenix'06 Annual Technical Conference (Boston, MA, May 30 – June 03, 2006). USENIX Association, Berkeley, CA, 31–31.
16. J. Liang, R. Kumar, K. Ross. The FastTrack Overlay: A Measurement Study. Computer Networks, 50, 2006, 842–858.
17. S. Baset and H. Schulzrinne. An Analysis of the Skype Peer-to-Peer Internet Telephony Protocol Columbia University, Department of Computer Science, Technical Report cucs-039-04. 2004.
18. S. Guha, N. Daswani, and R. Jain. An experimental study of the Skype peer-to-peer VoIP system. In IPTPS, 2006.
19. T. Hofeld. Measurement and Analysis of Skype VoIP Traffic in 3G UMTS Systems. 4th International Workshop on Internet Performance, Simulation, Monitoring and Measurement, IPS-MoMe 2006, Salzburg, Austria, February 2006.
20. K. Suh, D. R. Figueiredo, J. Kurose, and D. Towsley. Characterizing and detecting skype-relayed traffic, in Proceedings of IEEE Infocom (Infocom 2006), Barcelona, Spain, April, 2006.
21. H. Xie and Y. Yang. A Measurement-based Study of the Skype Peer-to-Peer VoIP Performance The Sixth International Workshop on Peer-to-Peer Systems. IPTPS 2007 (Feb 2007).
22. R. Schollmeier. A Definition of Peer-to-Peer Networking for the Classification of Peer-to-Peer Architectures and Applications. Peer-to-Peer Computing 2001.
23. S. Androutsellis-Theotokis and D. Spinellis. A Survey of Content Distribution Technologies. ACM Computing Surveys, 36(4), December 2004.
24. J. Buford. Management of peer-to-peer overlays, International Journal of Internet Protocol Technology, Special Issue on Management of IP Networks and Services, 3(1), 2008, 2–12.
25. B. Biskupski, J. Dowling, and J. Sacha. Properties and mechanisms of self-organizing MANET and P2P systems, ACM Transactions on Autonomous and Adaptive Systems 2(1), March 2007, 34 pp.
26. J. Buford, H. Yu, and E. K. Lua. P2P Networking and Applications. Morgan Kaufmann 2008.

Unstructured Overlays

27. I. Clarke. A Distributed Decentralized Information Storage and Retrieval System. Unpublished Report. Division of Informatics, University of Edinburgh. 1999. Online: http://www.freenetproject.org.

28. I. Clarke, O. Sandberg, B. Wiley, and T. W. Hong. Freenet: A distributed anonymous information storage and retrieval system, in Proc. ICSI Workshop Design Issues in Anonymity and Unobservability, Berkeley, CA, June 2000. (also: Hannes Federrath (Ed.): Designing Privacy Enhancing Technologies. International Workshop on Design Issues in Anonymity and Unobservability. Lecture Notes in Computer Science VOL. 2009, Springer-Verlag, Berlin/ Heidelberg 2001.
29. H.-E. Skogh, J. Haeggstrom, A. Ghodsi, and R. Ayani. Fast Freenet: Improving Freenet Performance by Preferential Partition Routing and File Mesh Propagation. In Proceedings of the 6th International Workshop on Global and Peer-To-Peer Computing on Large Scale Distributed Systems (CCGRID'06), p. 9. IEEE Computer Society, 2006.
30. H. Zhang, A. Goel, and R. Govindan. Using the small-world model to improve Freenet performance. Comput. Networks 46(4), Nov. 2004, 555–574.
31. Gnutella Protocol Specification version 0.6.
Online at http://gnutella-specs.rakjar.de/index.php/ Main_Page, accessed Nov 2007.
32. M. Ripeanu, I. Foster, and A. Iamnitchi. Mapping the Gnutella Network: Properties of Large-Scale Peer-to-Peer Systems and Implications for System Design, IEEE Internet Computing, 6(1), February 2002.
33. Y. Chawathe, S. Ratnasamy, L. Breslau, S. Shenker, and N. Lanham. GIA: Making Gnutella-like P2P Systems. Scalable. ACM SIGCOMM 2003.
34. J. Liang, R. Kumar, and K. Ross. The FastTrack Overlay: A Measurement Study. Computer Networks, 50, 2006, 842–858.
35. R. Morselli, B. Bhattacharjee, A. Srinivasan, and M. Marsh. Efficient lookup on unstructured topologies. Proceedings of the Twenty-Fourth Annual ACM Symposium on Principles of Distributed Computing (Las Vegas, NV, USA, July 17 – 20, 2005). PODC '05. ACM Press, New York, NY, 77–86.
36. K. Hui, J. Lui, and D. Yau. Small-world overlay P2P networks: construction, management and handling of dynamic flash crowds. Comput. Networks, 50(15), Oct. 2006, 2727–2746.
37. A. Löser, S. Staab, and C. Tempich. Semantic Social Overlay Networks. IEEE J. Sel. Areas. Communications, 25(1), 2007, 5–14.
38. J. Pouwelse, P. Garbacki, J. Wang, A. Bakker, J. Yang, A. Iosup, D. Epema, M. Reinders, M. van Steen, and H. Sips. Tribler: A Social-based Peer-to-Peer system. Proc. of the 5th International Workshop on Peer-to-Peer Systems (IPTPS'06).
39. G. Mangioni, V. Carchiolo, M. Malgeri, and V. Nicosia, Evaluating the Dynamic Behaviour of PROSA P2P Network, International Symposium on Parallel and Distributed Processing and Applications 2006, ISPA06, 2006.
40. P. Ganesan, Q. Sun, and H. Garcia-Molina. Yappers: A peer-to-peer lookup service over arbitray topology. Infocom'03, Apr. 2003.

Broadcast in Structured Overlays

41. M. Lue, C. King, and H. Fang. Scoped broadcast in structured P2P networks. Proc. of the 1st International Conference on Scalable information Systems (Hong Kong, May 30 – June 01, 2006). InfoScale '06, vol. 152. ACM Press, New York, NY, 51.
42. V. Vischnevsky, A. Safonov, M. Yakimov, E. Shim, and A. Gelman. Scalable Blind Search and Broadcasting in Peer-to-Peer Networks, Sixth IEEE Intl. Conference on Peer-to-Peer Computing, Sept. 2006.
43. J. Li, K. Sollins, and D. Lim. Implementing aggregation and broadcast over Distributed Hash Tables. SIGCOMM Computer Communication Review, 35(1), Jan. 2005, 81–92.
44. H.-C. Hsiao and C.-T. King. Scoped broadcast in dynamic peer-to-peer networks. 29th Annual International Computer Software and Applications Conference, 2005. COMPSAC 2005. 26–28 July 2005, pp. 533–538 Vol. 2.

Structured Overlays

45. C. Greg Plaxton, R. Rajaraman, and A. Richa. Accessing nearby copies of replicated objects in a distributed environment, Proceedings of the ninth annual ACM symposium on Parallel algorithms and architectures, p. 311–320, June 23–25, 1997, Newport, Rhode Island, United States.
46. X. Li, J. Misra, and C. G. Plaxton. Active and Concurrent Topology Maintenance. In Proceedings of the 18th International Conference on Distributed Computing (DISC 04), p. 320–334, London, UK, 2004. Springer-Verlag.
47. B. Y. Zhao, L. Huang, J. Stribling, S. C. Rhea, A. D. Joseph, and J. Kubiatowicz. Tapestry: A resilient global-scale overlay for service deployment, IEEE Journal on Selected Areas in Communications, 22(1), pp. 41–53, Jan. 2004.
48. A. Rowstron, P. Druschel. Pastry: Scalable, Decentralized Object Location, and Routing for Large-Scale Peer-to-Peer Systems, Proceedings of the IFIP/ACM International Conference on Distributed Systems Platforms Heidelberg, p.329–350, November 12–16, 2001
49. K. Aberer, A. Datta, M. Hauswirth. Efficient, self-contained handling of identity in Peer-to-Peer systems, IEEE Transactions on Knowledge and Data Engineering 16(7), July 2004.
50. S. Rhea, B. Godfrey, B. Karp, J. Kubiatowicz, S. Ratnasamy, S. Shenker, I. Stoica, and H. Yu. OpenDHT: A Public DHT Service and Its Uses. Proceedings of ACM SIGCOMM 2005, August 2005.
51. Q. Lian, Z. Zhang, S. Wu, and B. Zhao. Z-Ring: Fast Prefix Routing via a Low Maintenance Membership Protocol. In Proceedings of the 13TH IEEE international Conference on Network Protocols (Icnp'05) – Volume 00 (November 06–09, 2005). ICNP. IEEE Computer Society, Washington, DC, 132–146.
52. I. Stoica, R. Morris, D. Liben-Nowell, D. Karger, M. F. Kaashoek, F. Dabek, and H. Balakrishnan. Chord: a scalable peer-to-peer lookup protocol for internet applications. IEEE/ACM Transactions on Networking 11(1) Feb. 2003, 17–32.
53. L. Alima et al., Dks(n,k,f): A Family of Low Communication, Scalable and Fault-Tolerant Infrastructures for P2P Applications, Proc. 3rd IEEE/ACM Int'l. Symp. Cluster Comp. and the Grid, Monterey, California, USA, 2003, pp. 344–50.
54. B. Carton, V. Mesaros, and P. Van Roy. Improving the scalability of logarithmic-degree DHT-based peer-to-peer networks,Proc. of Euro-Par, Aug.-Sept. 2004.
55. T. Schutt, F. Schintke, and A. Reinefeld. Structured Overlay without Consistent Hashing: Empirical Results. In Proceedings of the Sixth IEEE international Symposium on Cluster Computing and the Grid (Ccgrid'06) – Volume 00 (May 16 – 19, 2006). CCGRID. IEEE Computer Society, Washington, DC, 8.
56. G. Manku, M. Bawa, and P. Raghavan. Symphony: distributed hashing in a small world. In Proceedings of the 4th Conference on USENIX Symposium on internet Technologies and Systems – Volume 4 (Seattle, WA, March 26 – 28, 2003). USENIX Association, Berkeley, CA, 10–10.
57. P. Maymounkov and D. Mazieres. Kademlia: A peer-to-peer information system based on the xor metric. In Proc of IPTPS02, Cambridge, USA, March 2002. 7.5 Constant Degree Overlays.
58. S. Ratnasamy, P. Francis, M. Handley, R. Karp, S. Schenker. A scalable content-addressable network, Proceedings of the 2001 conference on Applications, technologies, architectures, and protocols for computer communications, p.161–172, August 2001, San Diego, California.
59. D. Malkhi, M. Naor, and D. Ratajczak. Viceroy: a scalable and dynamic emulation of the butterfly. In Proceedings of the Twenty-First Annual Symposium on Principles of Distributed Computing (Monterey, California, July 21 – 24, 2002). PODC '02. ACM Press, New York, NY, 183–192.
60. A. Kumar, S. Merugu, J. Xu, and X. Yu. Ulysses: A Robust, Low-Diameter, Low-Latency Peer-to-peer Network, in Proc. of IEEE ICNP 2003.

61. H. Shen, C.-Z. Xu, and G. Chen. Cycloid: A scalable constant-degree lookup-efficient P2P overlay network, Journal of Performance Evaluation's Special Issue on Peer-to-Peer Networks (6/29), 2005.
62. C. Shui, H. Wang, P. Zhou, and Y. Jia. Cactus: A New Constant-Degree and Fault Tolerate P2P Overlay. PRIMA 2006: 386–397.
63. F. Kaashoek and D. R. Karger. Koorde: A Simple Degree-optimal Hash Table, IPTPS, February 2003.
64. A.-T. Gai and L. Viennot. Broose: a practical distributed hashtable based on the de-bruijn topology, in Fourth International Conference on Peer-to-Peer Computing, 2004, Aug. 2004, pp. 167–174.
65. P. Fraigniaud and P. Gauron. D2B: a de Bruijn based content-addressable network. Theor. Comput. Sci. 355, 1 (Apr. 2006), 65–79.
66. M. Naor and U. Wieder. Novel architectures for P2P applications: The continuous-discrete approach. ACM Trans. Algorithms 3, 3 (Aug. 2007), 34.
67. G. Wepiw and P. Simeonov. HiPeer: A Highly Reliable P2P System. IEICE – Transactions on Information and Systems, E89-D(2), Feb. 2006, 570–580.
68. H. Rostami, J. Habibi, and A. Rahnama. Semantic HyperCup. In Proceedings of the 39th Annual Hawaii international Conference on System Sciences – Volume 09 (January 04 – 07, 2006). HICSS. IEEE Computer Society, Washington, DC, 223.
69. D. Li, X. Lu, and J. Wu. FISSIONE: a scalable constant degree and low congestion DHT scheme based on Kautz graphs. Proc. IEEE 24th Annual Joint Conference of the IEEE Computer and Communications Societies (INFOCOM 2005), 3, 13–17 March 2005, 1677–1688.
70. D. Guo, J. Wu, H. Chen, and X. Luo. Moore: An Extendable Peer-to-Peer Network Based on Incomplete Kautz Digraph with Constant Degree. INFOCOM 2007.
71. Y. Zhang, X. Lu, and D. Li. SKY: efficient peer-to-peer networks based on distributed Kautz graphs. Journal Science in China Series F: Science in China Press, co-published with Springer-Verlag GmbH. Dec. 2008.
72. Y. Zhang, L. Liu, D. Li, and X. Lu. Distributed Line Graphs: A Universal Framework for Building DHTs Based on Arbitrary Constant-Degree Graphs. Proceedings of the 2008 the 28th international Conference on Distributed Computing Systems – Volume 00 (June 17 – 20, 2008). ICDCS. IEEE Computer Society, Washington, DC, 152–159.

O(1)-Hop Overlays

73. B. Leong, B. Liskov, and E. D. Demaine. EpiChord: Parallelizing the Chord Lookup Algorithm with Reactive Routing State Management. Computer Communications, Elsevier Science, 29, 2006, 1243–1259.
74. A. Gupta, B. Liskov, and R. Rodrigues. One Hop Lookups for peer-to-peer overlays Proceedings of the 9th Workshop on Hot Topics in Operating Systems (HotOS-IX), Lihue, Hawaii, May 2003.
75. I. Gupta, K. Birman, P. Linga, A. Demers, and R. van Renesse. Kelips: building an efficient and stable P2P DHT through increased memory and background overhead. Proceedings of the 2nd International Workshop on Peer-to-Peer Systems (IPTPS '03). 2003.
76. I. Abraham, A. Badola, D. Bickson, D. Malkhi, S. Maloo, and S. Ron. Practical Locality-Awareness for Large Scale Information Sharing. The 4th Annual International Workshop on Peer-To-Peer Systems (IPTPS '05). 2005.
77. M. Li, J. Hu, D. Wang, and W. Zheng. Gemini: Probabilistic Routing Algorithm in Structured P2P Overlay. The 3rd International Conference on Grid and Cooperative Computing (GCC 2004), Wuhan, China, Oct 2004.
78. L. Monnerat and C. Amorim. D1HT: A Distributed One Hop Hash Table. In Proc of the 20th IEEE Intl Parallel & Distributed Processing Symposium (IPDPS), April 2006. 7.7 Variable Hop Overlays.

79. J. Li, J. Stribling, R. Morris, and M. F. Kaashoek. Bandwidth-efficient Management of DHT Routing Tables, NSDI 2005.
80. A. Brown, M. Kolberg, and J. Buford. Chameleon: An adaptable 2-tier variable hop overlay. IEEE CCNC 2009, Jan. 2009.
81. A. Brown, J. Buford, and M. Kolberg. Tork: A Variable-Hop Overlay for Heterogeneous Networks. Fourth Workshop on Mobile Peer-to-Peer 2007. March 2007.

Hierarchical and Federated Overlays

82. M. Castro, P. Druschel, A.-M. Kermarrec, and A. Rowstron. One ring to rule them all: service discovery and binding in structured peer-to-peer overlay networks, Proc. of the 10th workshop on ACM SIGOPS European workshop: beyond the PC, July 01-01, 2002, Saint-Emilion, France.
83. L. Garces-Erce, K.W. Ross, E. Biersack, P. Felber, and G. Urvoy-Keller. TOPLUS: Topology Centric Lookup Service, Fifth International Workshop on Networked Group Communications (NGC'03), Munich, September 2003.
84. Z. Xu, R. Min, and Y. Hu. HIERAS: a DHT based hierarchical P2P routing algorithm, International Conference on Parallel Processing (ICPP'03), Kaohsiung, Taiwan, 2003.
85. M. Artigas , P. Lopez , J. Ahullo, and A. Skarmeta. Cyclone: A Novel Design Schema for Hierarchical DHTs, Proc. of the Fifth IEEE International Conference on Peer-to-Peer Computing, p.49–56, August 31-September 02, 2005.
86. Q. Lian, Z. Zhang, S. Wu, and B. Zhao. Z-Ring: Fast Prefix Routing via a Low Maintenance Membership Protocol. In Proceedings of the 13th IEEE international Conference on Network Protocols (Icnp'05) – Volume 00 (November 06 - 09, 2005). ICNP. IEEE Computer Society, Washington, DC, 132–146.
87. P. Ganesan, K. Gummadi, and H. Garcia-Molina. Canon in G Major: Designing DHTs with Hierarchical Structure, IEEE International Conference on Distributed Computing Systems (ICDCS 2004), Tokyo, Japan, pp. 263–272, 2004.
88. M. Freedman and D. Mazières. Sloppy Hashing and Self-Organizing Clusters. Proc. 2nd Intl. Workshop on Peer-to-Peer Systems (IPTPS '03) Berkeley, CA, February 2003.
89. D. Braun, J. Buford, et al. UP2P: A Peer-to-Peer, Overlay Architecture for Ubiquitous Communications and Networking. IEEE Communications Magazine. 12/2008.

Service Overlays

90. D. G. Andersen, H. Balakrishnan, M. F. Kaashoek, and R. Morris. Resilient overlay networks. Proc. 18th ACM Symposium on Operating Systems Principles (SOSP) (Banff, Canada, Oct. 2001), pp. 131–145.
91. D. Andersen, A. Snoeren, and H. Balakrishnan. Best-path vs. multi-path overlay routing. Proceedings of the 3rd ACM SIGCOMM conference on Internet measurement (IMC'03), 2003.
92. S. Qazi and T. Moors. Scalable Resilient Overlay Networks Using Destination-Guided Detouring. Proc. IEEE International Conference on Communications (ICC), Jun. 2007.
93. Y. Zhu, C. Dovrolis, and M. Ammar. Proactive and reactive bandwidth driven overlay routing: A simulation study. Computer Networks, 50(6), April 2006, 742–762.
94. S.-J. Lee, S. Banerjee, P. Sharma, P. Yalagandula, and S. Basu. Bandwidth-Aware Routing in Overlay Networks IEEE INFOCOM 2008. The 27th Conference on Computer Communications 13–18 April 2008, p. 1732–1740.
95. R. Cox, A. Muthitacharoen, and R. Morris. Serving DNS Using a Peer-to-Peer Lookup Service. In Revised Papers From the First international Workshop on Peer-To-Peer Systems

(March 07 – 08, 2002). P. Druschel, M. F. Kaashoek, and A. I. Rowstron, Eds. Lecture Notes In Computer Science, vol. 2429. Springer-Verlag, London, 155–165.
96. J. Jung, E. Sit, H. Balakrishnan, and R. Morris. DNS performance and the effectiveness of caching. Proceedings of the ACM SIGCOMM Internet Measurement Workshop '01, San Francisco, California, November 2001.
97. V. Pappas, D. Massey, A. Terzis, L. Zhang. A Comparative Study of Current DNS with DHT-Based Alternatives. IEEE INFOCOM 2006, April 2006.
98. V. Ramasubramanian and E. Sirer. The design and implementation of a next generation name service for the internet. In Proceedings of the 2004 Conference on Applications, Technologies, Architectures, and Protocols For Computer Communications (Portland, Oregon, USA, August 30 – September 03, 2004). SIGCOMM '04. ACM, New York, NY, 331–342.

Sensor Overlays

99. PIAX. piax.org

Churn and Overlay Maintenance

100. D. Stutzbach and R. Rejaie. Understanding churn in peer-to-peer networks. In Proceedings of the 6th ACM SIGCOMM Conference on Internet Measurement Conference 2006 (IMC 2006), October 25–27, 2006.
101. S. Guha, N. Daswani, and R. Jain. An experimental study of the Skype peer-to-peer VoIP system, 5th Annual International Workshop on Peer-to-Peer Systems (IPTPS'06), 2006.
102. P. Kersch, R. Szabo, L. Cheng, K. Jean, and A. Galis. Stochastic maintenance of overlays in structured P2P systems. Elsevier Journal of Computer Communications, Special Issue: Disruptive networking with peer-to-peer systems, 31(3), 25 Feb. 2008, 603–619.
103. D. Wu, Y. Tian, K.-W. Ng and A. Datta. Stochastic analysis of the interplay between object maintenance and churn. Elsevier Journal of Computer Communications, 31(3), Feb. 2008.

Mobile P2P

104. A. MacQuire, A. Brampton, I. Rai and L. Mathy. Performance Analysis of Stealth DHT with Mobile Nodes. In Proceedings of the 4th IEEE International Conference on Pervasive Computing and Communications Workshops (PERCOMW 2006), March 2006.
105. H.-C. Hsiao and C.-T. King. Bristle: A Mobile Structured Peer-to-Peer Architecture. International Parallel and Distributed Processing Symposium (IPDPS'03), 2003. 7.13 Overlays for MANETs.
106. O. Landsiedel, S. Götz, and K. Wehrle. Towards Scalable Mobility in Distributed Hash Tables. Sixth International IEEE Conference on Peer-to-Peer-Computing, Cambridge, UK, August / September 2006.
107. H. Pucha, S. Das, and Y. C. Hu. Ekta: An Efficient DHT Substrate for Distributed Applications in Mobile Ad Hoc Networks. Proc. 6th IEEE Workshop on Mobile Computing Systems and Applications (WMCSA), 2004, pp. 163–173.
108. I. Gruber, R. Schollmeier, and W. Kellerer. Performance evaluation of the mobile peer-to-peer service. Proceedings IEEE CCGrid 2004, 2004, pp. 363–371.
109. M. Conti, E. Gregori, and G. Turi. A crosslayer optimization of gnutella for mobile ad hoc networks. Proc. of the 6th ACM international symposium on Mobile ad hoc networking and computing (MobiHoc05), pp. 343–354, Urbana-Champaign, IL, USA, 2005. ACM Press.

110. T. Zahn and J. Schiller. Designing Structured Peer-to-Peer Overlays as a Platform for Distributed Network Applications in Mobile Ad Hoc Networks. Compute Communications, 31(2), 5 February 2008, 643–654.
111. B. Tang, Z. Zhou, A. Kashyap, and T. Chiueh. An Integrated Approach for P2P File Sharing on Multi-hopWireless Networks. In Proc. of the IEEE Int. Conf. on Wireless and Mobile Computing, Networking and Communication (WIMOB'05), Montreal (Canada), August 2005.
112. A. Klemm, C. Lindemann, and O. Waldhorst. A special-purpose peer-to-peer file sharing system for mobile ad hoc networks. Proceeding Workshop on Mobile Ad Hoc Networking and Computing (MADNET 2003), 2003, 41–49
113. C. Cramer and T. Fuhrmann. ISPRP: A Message-Efficient Protocol for Initializing Structured P2P Networks. Proc. 24th IEEE International Performance, Computing, and Communications Conference (IPCCC 2005), 2005, pp. 365–370.
114. H. Pucha, S. M. Das, and Y. C. Hu. Imposed Route Reuse in Ad Hoc Network Routing Protocols Using Structured Peer-to-Peer Overlay Routing. IEEE Transactions on Parallel Distributed Systems, 17(12), Dec. 2006, 1452–1467. 7.14 Variable Hop Overlays
115. J. Li, J. Stribling, R. Morris, and M. F. Kaashoek. Bandwidth-efficient management of DHT routing tables. In the Proceedings of the 2nd USENIX Symposium on Networked Systems Design and Implementation (NSDI '05), Boston, MA, 2005.
116. A. Brown, J. Buford, and M. Kolberg. Tork: A Variable-Hop Overlay for Heterogeneous Networks. Fourth Workshop on Mobile Peer-to-Peer 2007. March 2007.
117. A. Brown, M. Kolberg, and J. Buford. An Adaptable Service Overlay for Wide-Area Network Service Discovery. IEEE Globecom 2007 Workshop – Enabling the Future Service-Oriented Internet. Nov. 2007.
118. A. Brown, M. Kolberg, and J. Buford. Chameleon: An adaptable 2-tier variable hop overlay. IEEE CCNC 2009, Jan. 2009.
119. J. Buford. Mobile P2P after Five Years – Where are we and where are we headed? (Contributed Talk) Fifth IEEE Workshop on Mobile Peer-to-Peer. March 2008.

Content Access and Delivery

120. K. Yang and J. Ho, Proof: A DHT-Based Peer-to-Peer Search Engine, In Proceedings the IEEE International Conference on Web Intelligence, 2006, Hong Kong: IEEE Computer Society, p. 702–708.
121. B. Yang and H. Garcia-Molina. Efficient search in peer-to-peer networks, in Proceedings the 22nd International Conference on Distributed Computing, 2002.
122. V. Kalogeraki, D. Gunopulos, and D. Zeinalipour-yazti. A local search mechanism for peer-to-peer networks, in Proceedings of the Eleventh ACM Conference on Information and Knowledge Management, 2002.
123. G. Gupta, D. Agrawal, and A. E. Abbadi. "Approximate range selection queries in peer-to-peer systems," in Proceedings of the First Biennial Conference on Innovative Data Systems Research, Asilomar CA, USA, 2003.
124. C. Zhang, A. Krishnamurthy, and R. Y. Wang. "SkipIndex: Towards a Scalable Peer-to-Peer Index Service for High Dimensional Data." Available at http://www.cs.princeton.edu/chizhang/skipindex.pdf
125. M. Castro, P. Druschel, A.-M. Kermarrec, A. Nandi, A. Rowstron, and A. Singh. "Splitstream: Highbandwidth multicast in cooperative environments," in Proceedings, the 20th ACM Symp. on Operating Sys. Principles (SOSP 2003), Oct. 2003.
126. H. Yu and J. Buford. Peer-to-peer overlay multicast, Book Chapter, in Encyclopedia of Wireless and Mobile Communications, Auerbach Publications, Florida, USA, to appear November, 2007.

127. S. Banerjee, B. Bhattacharjee, and C. Kommareddy. Scalable application layer multicast, in Proceedings, ACM SIGCOMM2002, September 2002.

Security

128. Web Application Security Consortium, Web Application Security Consortium: Threat Classification, Version 1.00, 2004. Available at http://www.webappsec.org/projects/threat/
129. S. Hansman and R. Hunt. A taxonomy of network and computer attacks, Computers & Security, 24(1), 2005, 31–43.
130. A. Simmonds, P. Sandilands, and L. van Ekert. An ontology for network security attacks, International Journal of Information Security and Privacy, 1(4), 2007, 1–23.
131. D. Wallach. A survey of peer-to-peer security issues, in Proceedings, International Symposium on Software Security - Theories and Systems, Tokyo, Japan, November 2002, p.42–57.
132. M. Castro, P. Druschel, A. Ganesh, A. Rowstron, and D. S. Wallach. Secure routing for structured peer-to-peer overlay networks, in Proceedings of 5th Symp. on Operating Sys. Design and Impl., Boston, MA, December 2002, 299–314.
133. J. Mirkovic and P. Reiher. A taxonomy of DDoS attack and DDoS defense mechanisms, Computer Communications Review, 34(2), April 2004.
134. N. Naoumov and K. Ross. Exploiting P2P systems for DDoS attacks, in Proceedings of the 1st international conference on Scalable information systems, Hong Kong, 2006.
135. E. Adar and B. Huberman, Free riding on Gnutella, FirstMonday, 2000.

Surveys

136. S. Androutsellis-Theotokis and D. Spinellis. A Survey of Content Distribution Technologies. ACM Computing Surveys, 36(4), December 2004.
137. E. K. Lua, J. Crowcroft, M. Pias, R. Sharma, S. Lim. A Survey and Comparison of Peer-to-Peer Overlay Network Schemes. IEEE Communications Surveys and Tutorials, Second Quarter, 7(2), 2005.
138. J. Risson and T. Moors. Survey of research towards robust peer-to-peer networks: search methods. Computer Networks, 50(17), Dec. 2006, 3485–3521.
139. S. El-Ansary, S. Haridi. An Overview of Structured P2P Overlay Networks. Handbook on Theoretical and Algorithmic Aspects of Sensor, Ad Hoc Wireless, and Peer-to-Peer Networks (ed. J. Wu). Auerbach Publications, 2006, 665–683.
140. R. Steinmetz and K. Wehrle (Eds.). Peer-to-Peer Systems and Applications. Lecture Notes in Computer Science 3485 Springer 2005.