A Survey of Peer-to-Peer Content Distribution Technologies

Stephanos Androutsellis-Theotokis and Diomidis Spinellis

ACM Computing Surveys, December 2004

Wooyoung Park, Seyoung Kim

Defining Peer-to-Peer Computing

- Strictest definition of peer-to-peer
  - All nodes are completely equivalent in terms of functionality and tasks they perform.
  - These definitions fail to encompass systems that employ the notion of “supernodes”, which are widely accepted as peer-to-peer
- Definition in Shirky[2000]
  - “peer-to-peer is a class of applications that take advantage of resources—storage, cycles, content, human presence—available at the edges of the internet”
  - This definition encompass systems that completely rely upon centralized servers for their operation
- Author’s suggestion
  - Sharing of computer resources by direct exchange, rather than requiring the intermediation of a centralized server.
  - Ability to treat instability and variable connectivity, automatically adapting to failures in both network connections and computers

Author’s Definition Proposal

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

Classification of Peer-to-Peer Applications

- Communication and collaboration
  - Provide the infrastructure for facilitating direct, usually real time communication between peer computers.
- Distributed computation
  - Aim is to take advantage of the available peer computer processing power.
  - Breaking up and distributing the tasks and collecting the results.
- Internet Service Support
  - Multicast systems, security applications
- Database Systems
  - Local relation model : all data stored in inconsistent local relational databases.
- Content distribution
  - Include systems and infrastructures designed for the sharing of data between users
  - Relatively simple to more sophisticated systems
  - Napster, Gnutella, Freenet, Chord, Scan, PAST
  - This survey focus on content distribution

Peer-to-Peer Content Distribution

- Classification of P2P systems.
  - File exchange systems
    - Simple, one-off file exchanges between peers.
    - Light-weight without addressing security, availability, persistence.
  - P2P content publishing and storage systems
    - Distributed storage medium : users able to publish, store, and distribute content with security, persistency.
    - Security and persistency are main focus.
- Classification of P2P infrastructures
  - Routing and location
    - Route with efficiency and fault tolerance, efficiently locating peers and content
  - Anonymity
    - Provide user anonymity
  - Reputation management
    - Reputation update with various network nodes. Need reputation management infrastructures.

Analysis Framework

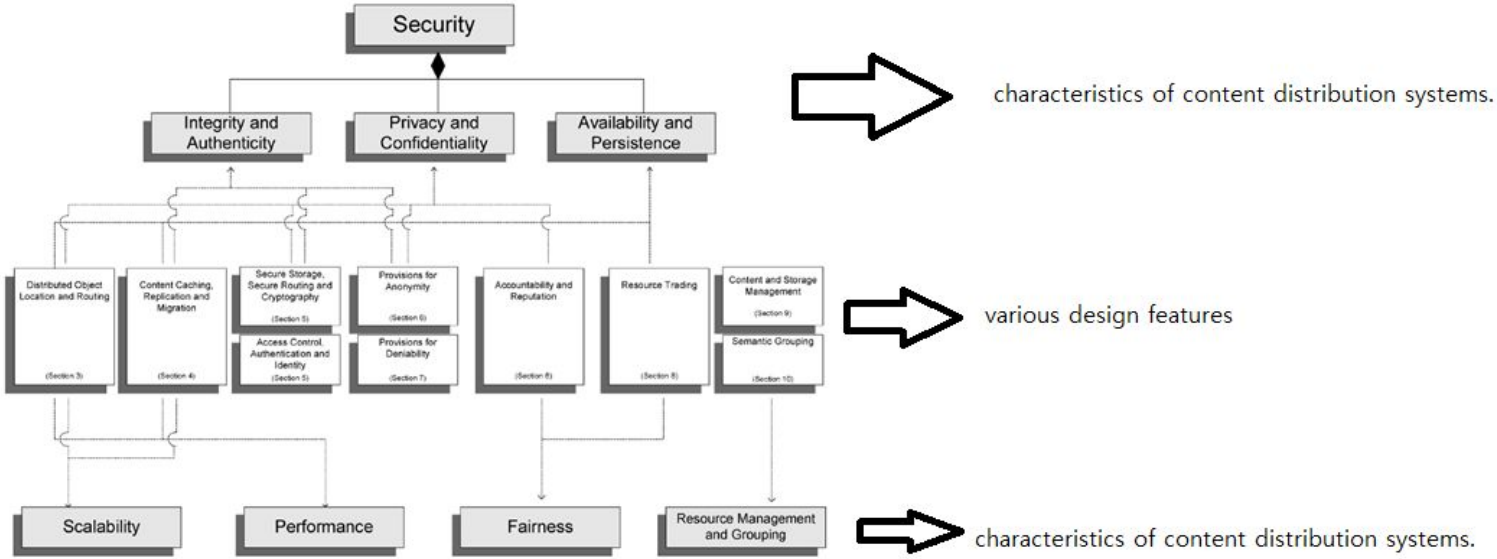


Fig. 1. Illustration of the way in which various design features affect the main characteristics of peer-to-peer content distribution systems.

Content Distribution System Characteristics

- Security
  - Integrity and authenticity : data completeness and safeguarding.
  - Privacy and confidentiality : only authorized user can access data.
  - Availability and persistence : authorized users have access to data when required.
- Scalability
  - Although the number of nodes/documents is dramatic increased, system’s performance should be maintained.
- Performance
- Fairness
- Resource management : management of editing, removal, storage space, metadata operation.

Peer-to-Peer Distributed Object Location and Routing

- Overlay network : network of peer computers and connections between them, formed on top of underlying physical computer network.
- Classify : centralization, structure.

Overlay Network Centralization

- Purely decentralized architectures
  - All nodes act as both servers and clients(servents)
- Partially centralized architectures
  - Some nodes has more important role(supernode)
- Hybrid decentralized architectures
  - Central server with querys.
  - unscalable

Network Structure

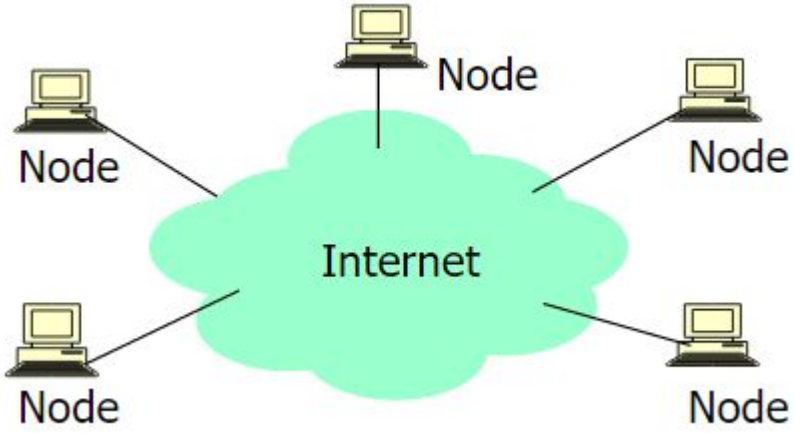
- Whether the overlay network is created deterministically or non-deterministically
- Unstructured
  - Location of file is completely unrelated to overlay topology.
  - Appropriate highly transient node populations.
  - Have matters of scalability, persistence, availability.
- Loosely structured
  - Location of file is affected by routing hints.
- Structured
  - File is placed at specified locations.
  - Offer exact-match queries.
  - Strong for scalability but week for highly transient node.

Classification of P2P Content Distribution

	Hybrid	Partial	none
Unstructured	Napster	Kazza	Gnutella
Loosely structured			Freenet
Structured			Chord, CAN, Tapestry

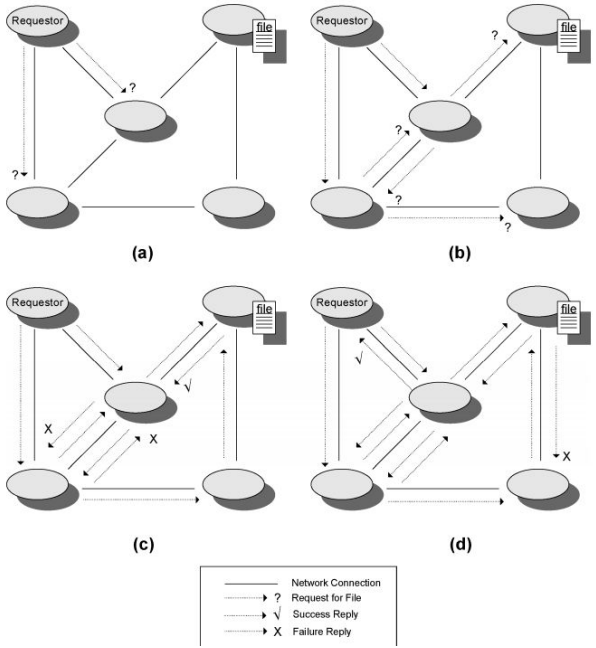
Unstructured architectures

- Purely decentralized architecture
  - All nodes perform exactly same tasks, act as both servers and clients(servents)
  - No central coordination
  - Users connect to each other directly through software application.
  - e.g. original Gnutella



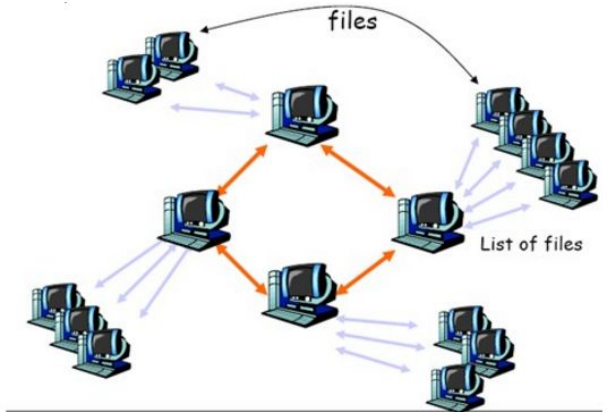
Purely Decentralized P2P Network

- Ping : request for connect
- Pong : reply message with host info
- Query : search request with
  search string and minimum speed req.
- Query hits : reply to query message
  with IP, port, speed.



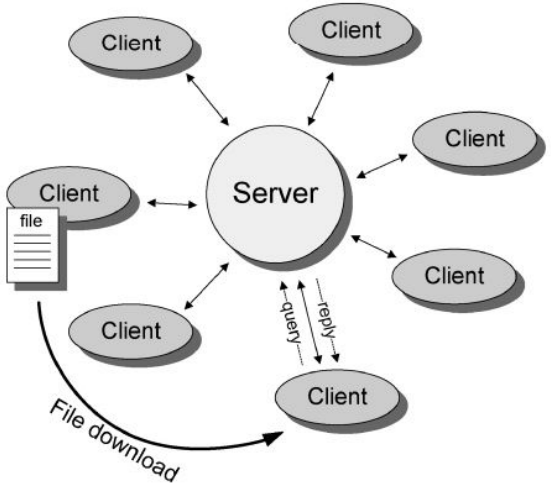
Unstructured Architectures

- Partially centralized architectures
  - Some nodes act as local central indexes for shared files(supernode)
  - Supernodes are dynamically assigned.
    Supernodes have connection with other supernodes.
  - Node with enough CPU power joins the network, it becomes supernode.
  - Advantage
    - Discovery time is reduced
    - Most of nodes(normal node) are lightly loaded
  - E.g. Kazza



Unstructured Architectures

- Hybrid decentralized architectures
  - E.g. Napster
  - Central server interacts between peers
  - Central server holds
    - directories of metadata(location of shared file)
    - User connection information table
  - Client gets file location from server, connect to client.
  - Advantage
    - Simple to implement
    - Locate file quickly and efficiently
  - Disadvantage
    - Vulnerable to attack, failure.
    - Lack of scalibility



Shortcoming and its Overcoming of Unstructured Architectures

- Unstructured P2P systems have unscalability matter.
  - Each node chooses a neighbor at random, propagates the request only to it [Lv et al 02]
  - Chooses a neighbor based on their past history, local indices. [Yang 02]
  - Choose a neighbor based on profile that have information about their performance in recent queries[Kalogeraki 02]

Freenet - Loosely Structured Architecture

- Loosely structured
  - Nodes can produce estimation of which nodes most likely to have files.
- use a **chain mode propagation**
  - Local decision of node to send
  - Avoid blindly broadcasting request message to all
- Files are identified by unique binary keys.
  - Made by hash function(simplest way)
- Nodes maintain local data store and dynamic routing table
- Freenet message form
  - Node ID, TTL, source and dest node ID

Freenet - Message Types

- Data insert
  - Inserts new data(key and actual data) in the network.
  - Insert process
    - Node calculates a binary key for the file.
    - Node sends a data insert message to itself.
    - If key is not found
      - Node looks up the closest key in its routing table, and forwards message to the corresponding node.
      - Newly inserted files are placed at nodes possessing files with similar keys.
    - If key is found
      - Node returns the preexisting file

Freenet - Message Types

- Data request
  - Request of the file using key of the file
  - Request process
    - If the node store a file
      - Search stop, forward back data to requestor with “data reply” message
    - If the node does not store a file
      - Forwards the request to its neighbor that is most likely to have the file
      - Message propagate from node-to-node
    - If hops-to-live value expired
      - Send back “data failed” message.

Freenet - Message Type

- Data reply
  - When the requested file is located, actual files with “data reply” message are sent to request node
- Data fail
  - When a node receives a “data fail” message, selects the next best node and forwards the request.
  - If all nodes have been explored and failed, it sends back “data fail” message.
- Data request/reply/fail procedure is backtracking search with limited number of forwarding.

Freenet - Indirect Files, Properties

- Indirect Files
  - A special class of lightweight files
  - Named according to search keywords
  - Contain pointers to the real file
  - Multiple files with the same key
- Freenet properties
  - Nodes specialize in searching for similar keys over time
  - Nodes store similar keys over time
  - Similarity of keys does not mean similarity of files.
  - Routing does not reflect the underlying network topology

Chord - Structured Infrastructures

- Identifies data items (files) with keys and storing the (key, data item) pairs at the node that the keys map to
- Keys are assigned both to files and nodes by means of a deterministic function.

Chord - Structured Infrastructures

- Identifier circle modulo 2^m
- Key k is assigned to node k or

  after k(the successor node of key k)
- Roughly the same number of keys
- Routing information : Each node should

  be aware of its successor node.

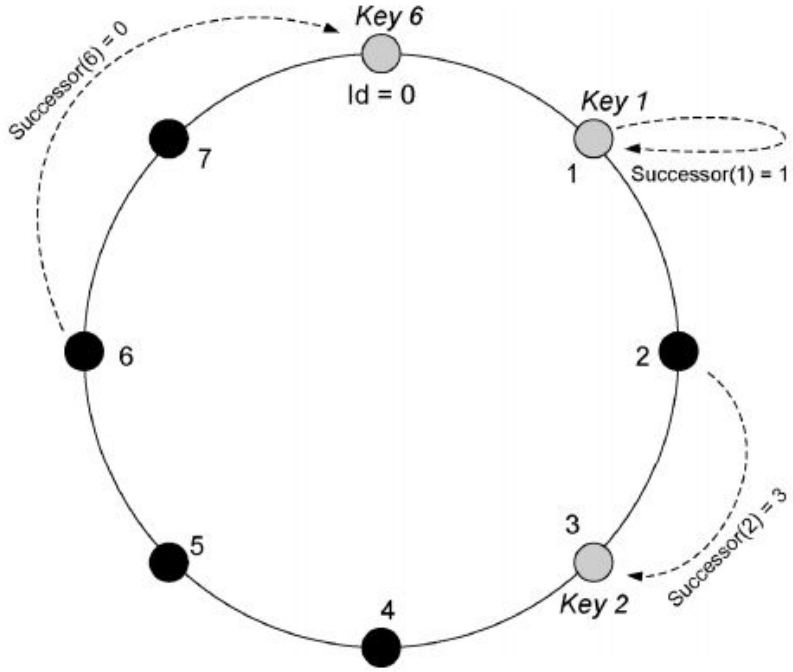


Fig. 5. A Chord identifier circle consisting of the three nodes 0,1 and 3. In this example, key 1 is located at node 1, key 2 at node 3, and key 6 at node 0.

Chord - Structured Infrastructures

- Queries for a given key are passed around the circle via these successor pointers until a node with the key.
- When a new node n joins, keys moves from n’s successor to node n.
- When node n leaves, all keys assigned to it will be reassigned to its successor.

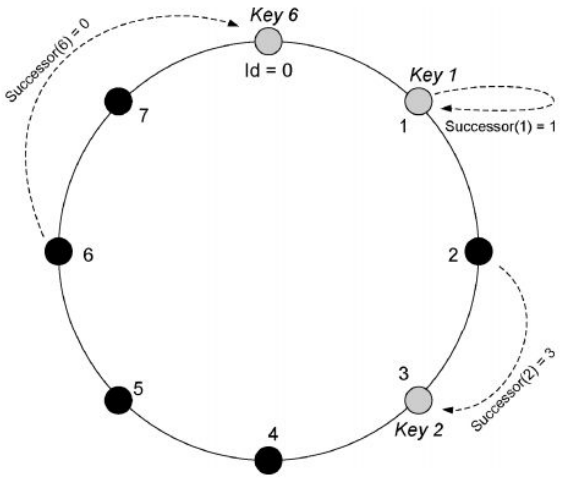


Fig. 5. A Chord identifier circle consisting of the three nodes 0,1 and 3. In this example, key 1 is located at node 1, key 2 at node 3, and key 6 at node 0.

Chord - Structured Infrastructures

- Only one data element per node need to be correct to guarantee correct routing of queries
- Availability remains high only as long as nodes fail independently
- Performance degrades gracefully

Chord - “finger table”

- In the worst case, Chord visits all N nodes to find a certain key.
- Chord uses “finger table” to increase the efficiency.

- Each entry i points to the successor of node n + 2i.
- For a node n to perform a lookup for key k, it identifies the highest node n* whose ID is between n and k.(if not found, the successor of n is returned)
- With the finger table, O(logN) bits & time.

4. Content Caching, Replication and Migration

- P2P content distribution systems rely on the replication of content for
- 1. The availability of content
- 2. Performance
- 3. Resisting censorship attempts

4. Content Caching, Replication and Migration

- Content replication can be categorized as follows:
  - Passive Replication
  - Cache-Based Replication
  - Active Replication
  - Instrospective replica management
  - Dynamic replica management

4.1. Passive Replication

- Content replication occurs naturally as nodes request and copy content from one another

4.2. Cache-Based Replication

- The result of caching copies of content as it passes through nodes
- In Freenet, a file is transferred through the network node-by-node
- In the process, copies of the file are cached on all intermediated nodes

4.3. Active Replication

- Proactive content replication and migration methods are often employed

4. Replication Issues

- By replicating content, data consistency and synchronization issues come up
- Some application effectively decide to weaken their consistency restrictions in favor of more extensive data replication and higher availability

5. Security

- Due to their open and autonomous nature, there are challenges:
  - Availability
  - Privacy
  - Confidentiality
  - Integrity
  - Authenticity
- The network nodes must be considered untrusted parties

5.1. Security - Secure Storage

- Self-Certifying Data
  - Integrity can be verified by the node retrieving it
- Information Dispersal
  - By Rabin[1989]
  - Files are encoded into m blocks
  - Any n is sufficient to reassemble the original data (m < n)
  - Gives resilience
- Distributed steganographic file systems
  - Encrypted blocks are indistinguishable from a random block
  - Their presence cannot even be detected

5.2. Security - Secure Routing

- The goal of secure routing : To resolve the problem of malicious nodes attempting to corrupt, delete, deny access to, or supply stale copies of object replicas that are transferred between nodes

5.3. Security - Access Control, Authentication and Identity Management

- The goal of secure routing : To resolve the problem of malicious nodes attempting to corrupt, delete, deny access to, or supply stale copies of object replicas that are transferred between nodes

5.3. Security - Access Control, Authentication and Identity Management

- “Sybil Attack” - same physical entity can appear under different identities
- It poses a security threat, especially in p2p systems that employ content replication or fragmentation schemes
- Douceur [2002], concludes that unless a central certification or identification authority is employed, p2p system will be susceptible to this kind of attack.
- The only proposed alternative: resource-demanding identification challenges

6. Provisions for Anonymity

- For privacy, confidentiality, and censorship resistance
- Anonymity refers to
  - The author of the content
  - The identity of a node storing the content
  - The identity and details of the content itself
  - The details of a query for retrieval of the content

6.1. Disassociation of Content Source and Requestor

- Freenet is designed to provide anonymity by
  - Making it infeasible to discover the true origin or destination of a file passing through its network
  - Making it difficult for a node operator to determine if responsible for the actual physical content of their own node
  - For anonymity, while a file is being passed, any node along this path can unilaterally decide to claim itself or another arbitrarily chosen node as the data source.

6.2. Anonymous Connection Layers

- Freenet is designed to provide anonymity by
  - Making it infeasible to discover the true origin or destination of a file passing through its network
  - Making it difficult for a node to determine if responsible for the actual physical content of their own node

6.3. Censorship Resistant Lookup

- Achord system proposed a censorship resistant design
- Publisher, storer, and retriever anonymity
- Difficult for a node to voluntarily assume responsibility for a certain document

7. Provisions for Deniability

- Refers to each user’s ability to deny knowledge of content stored in their node
- Users cannot be held responsible for the content stored in their node

7.1. Deniability of Stored Content

- Storing encrypted shares of files and no keys for them
  - Cannot know the content of the files whose shares they are storing
- Distributed steganographic storage systems
  - Offers deniability through the fact that blocks of files that are written on a peer node’s file system are undetectable

7.2. Deniability of Content in Transit

- The use of anonymous connection layers

7. Nondeniability in Structured Systems

- Structured systems are nondeniable.
- From a file, the identity of the node that stores it is known
- The owner of the node has no control over whether the file will be stored.
- That is, not responsible for its file

8. Incentive Mechanisms and Accountability

- An uncontrolled decentralized p2p system relies on the voluntary participation of its users.
- Necessary to provide incentives
- Operations, performance and availability rely on it

8.1. Reputation Mechanisms

- eBay - successful centralized reputation systems
- Goal is to take the locally generated reputation as a result of an interaction and spread it throughout the network to produce a global reputation
- The Eigen-Trust algorithm : global reputation based on the history of uploads
- The global reputation values are computed from the local reputation values assigned to a peer by other peers, weighted by the global reputation of the assigning peers

8.2. Micropayments Mechanisms

- In MojoNation, a currency(“Mojo”) is gained by offering disk space, bandwidth, or CPU cycles.
- Mojo is used to obtain access to distributed storage space.

8.3. Resource Trading Schemes

- Each node publishes and digitally signs logs containing
  - lists of files it stores on behalf of remote nodes
  - lists of files that other nodes are storing on its behalf
  - no node is using more resources than it is providing

- In FreeHaven, nodes form contracts to store each other’s material for a certain period of time
  - Fulfilling a contract -> increase in reputation
  - Dropping data earlier -> decrease in reputation

9. Resource Management Capabilities

- The minimum operations
  - Inserting
  - Locating
  - Retrieving
- Additional resource management facilities
  - Removing
  - Updating
  - Maintaining previous versions of updated content.

9.1. Content Deletion and Update

- Not straightforward
- MojoNation uses immutable files
- PAST offers a delete functionality for reclaiming the disc space occupied by a file
    - It does not guarantee that the file will no longer be available anywhere in the network

9.2. Storage and Bandwidth Management

- In PAST, users can use as much as they contribute on their node(quota system)
- To protect from denial of service attacks, apart from enforcing a published size quota to users
  - Requests a publisher to carry out computational work

9.3. Other Resource Management

- Content Versioning
- Directory Structure
- Content Searching

10. Semantic Grouping of Information

- Interest-based “peer communities”
  - Making searching more efficient
- The semantic overlay clustering approach
  - Based on partially-centralized(super-peer) networks
  - Creating logical layers above the physical network topology by matching semantic information provided by peers