RESEARCH ISSUES IN PEER-TO-PEER DATA MANAGEMENT

Özgür Ulusoy
Bilkent University

OUTLINE

- P2P computing systems
- Representative P2P systems
- P2P data management
- Incentive mechanisms
- Concluding remarks

PEER-TO-PEER COMPUTING

- Peer-to-Peer (P2P) is a distributed computing paradigm that enables a collection of nodes (peers) to share computer resources in a decentralized manner.
- Communication is symmetric; i.e., peers act as both clients and servers.

PEER-TO-PEER SYSTEMS

- Benefits of P2P model:
  - *self-organization*
  - *adaptation*
  - *scalability*
  - *autonomy*
  - *load-balancing*
  - *fault-tolerance*
  - *availability*
  - *resource aggregation*

SOME P2P STATISTICS

- At its peak, about 60 million people joined the *Napster* community. It managed to reach a peak population of approximately 1.5 million simultaneous users. [http://www.slyck.com/story1314.html]

- The number of simultaneous users participating on various P2P Networks increased from 3.8 million (Aug. 2003) to 9.0 million (Sept. 2006). [http://www.slyck.com/story1314.html]

- P2P population in Japan has increased from 1.3 million (2005) 1.8 million (2006) individuals. [http://www.slyck.com/story1249.html]

- P2P traffic in Germany consumes approximately 30% of available bandwidth during the day, and 70% at night. [http://www.slyck.com/story1320.html]

P2P DATA MANAGEMENT

- Initial motivation of P2P systems: file-sharing.
- To support sharing of semantically rich data, some data management issues need to be addressed:
  - data integration
  - query processing
  - data consistency
  - replication
  - clustering

P2P DATA MANAGEMENT

- Data management in P2P systems is challenging due to the large scale of the network and highly-transient peer population.
- Decentralized and self-adaptive data management is required.

P2P NETWORK STRUCTURE

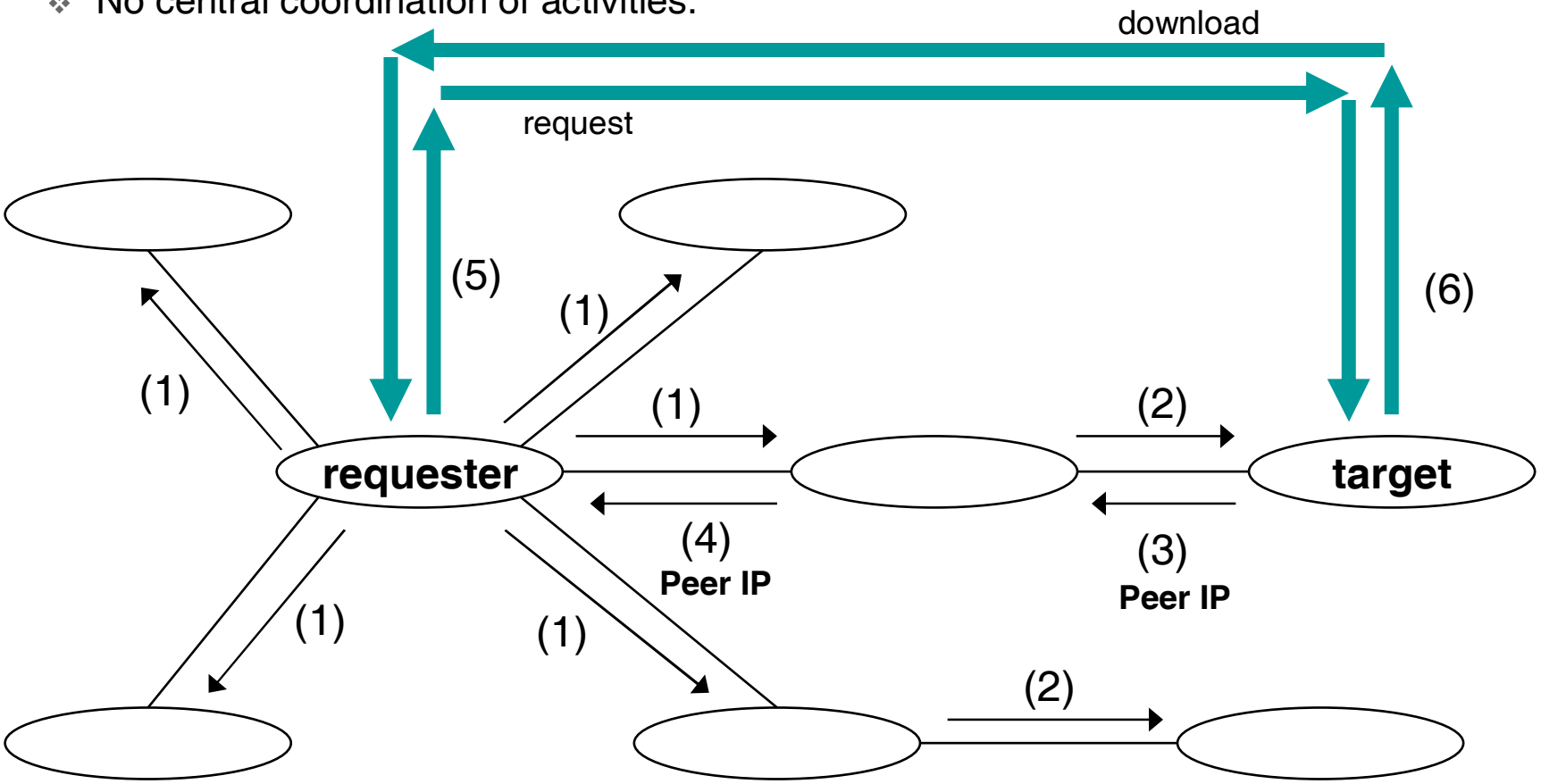
- Each peer maintains links with a selected subset of other peers, forming an *overlay network*.
- A message between any two peers is routed through the overlay network.
- Overlay networks can be distinguished in terms of their centralization and structure.

OVERLAY NETWORK CENTRALIZATION

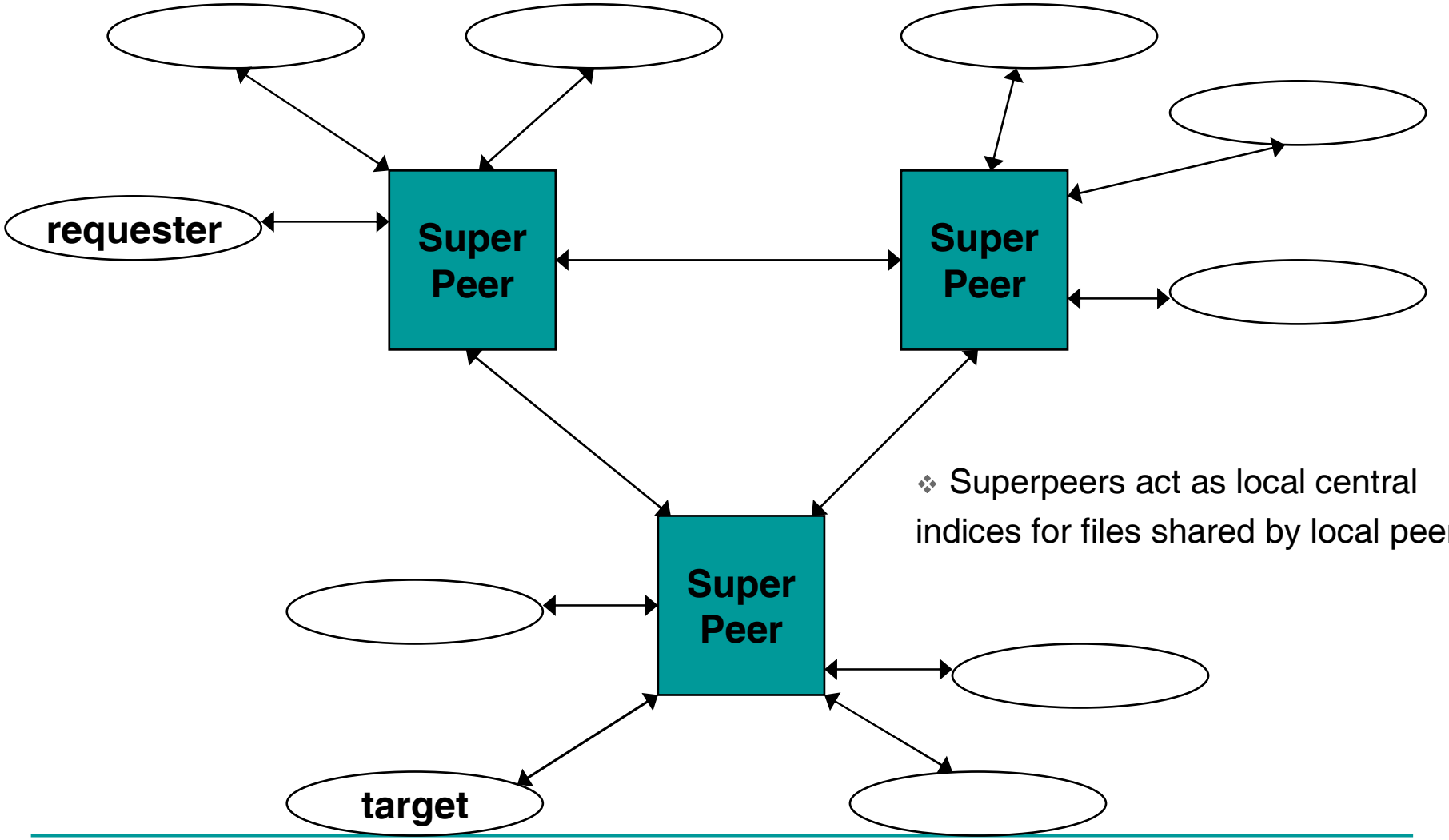
- *Purely Decentralized Architectures*
- *Partially Centralized Architectures*
- *Hybrid Decentralized Architectures*

Purely Decentralized

- No central coordination of activities.



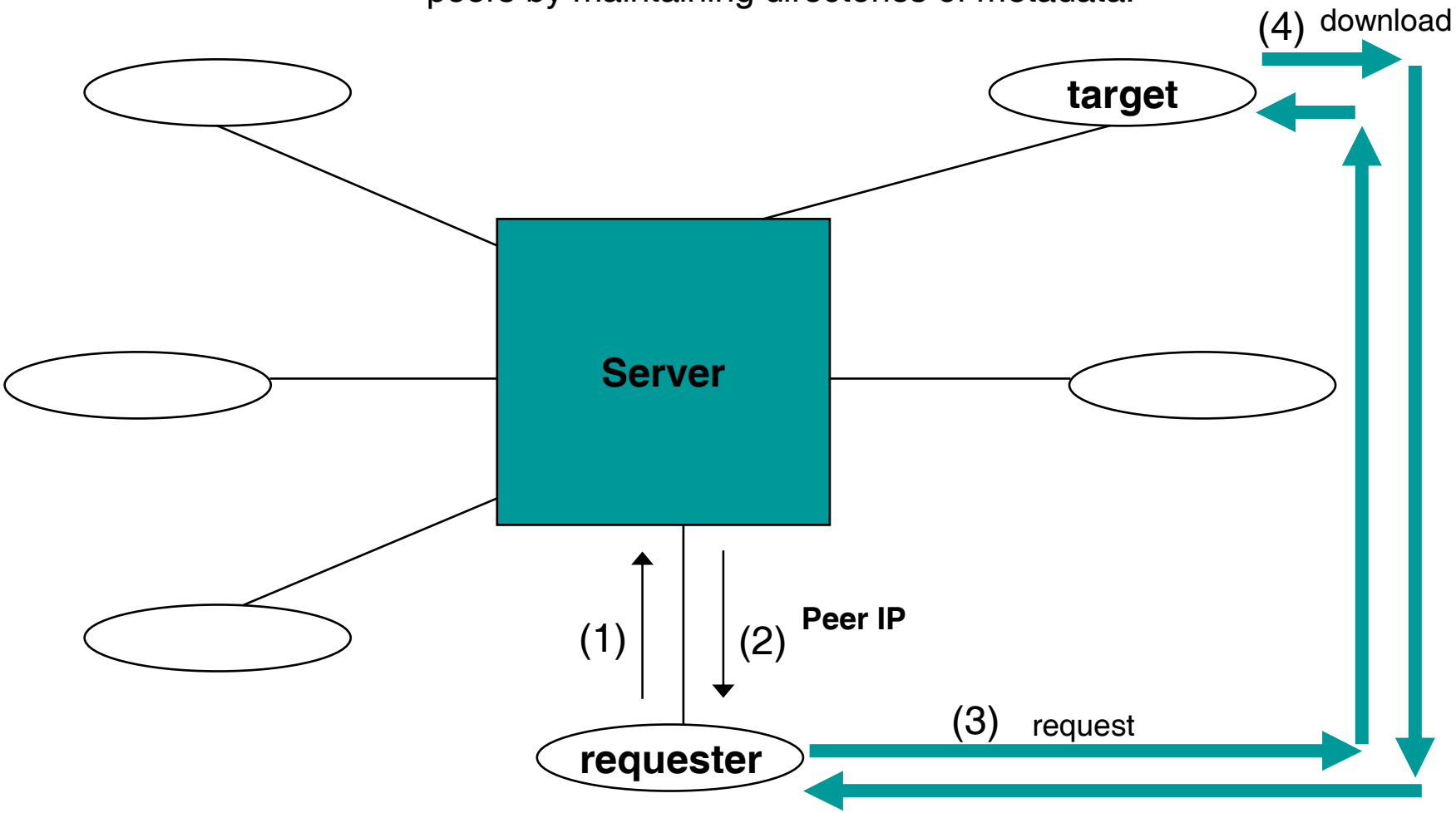
Partially Centralized



- Superpeers act as local central indices for files shared by local peers.

Hybrid Decentralized

A central server facilitates the interaction between peers by maintaining directories of metadata.



OVERLAY NETWORK STRUCTURE

Classification of overlay networks in terms of *structure*:

- *Unstructured*: overlay network is created nondeterministically as peers and files are added.
- *Structured*: creation of overlay networks is based on specific rules.

UNSTRUCTURED P2P SYSTEMS

- There is no restriction on data placement.
- Searching mechanisms employed can range from flooding to those that use indexing.
- Topologies are usually not restricted to some regular structure.
- Unstructured systems are more appropriate for accommodating highly-transient peer populations.

STRUCTURED P2P SYSTEMS

- Overlay topology is tightly controlled and pointers to files are placed at precisely specified locations.
- A mapping is provided between files and location in the form of a distributed routing table.
- It is hard to maintain the structure in the face of a highly-transient peer population.

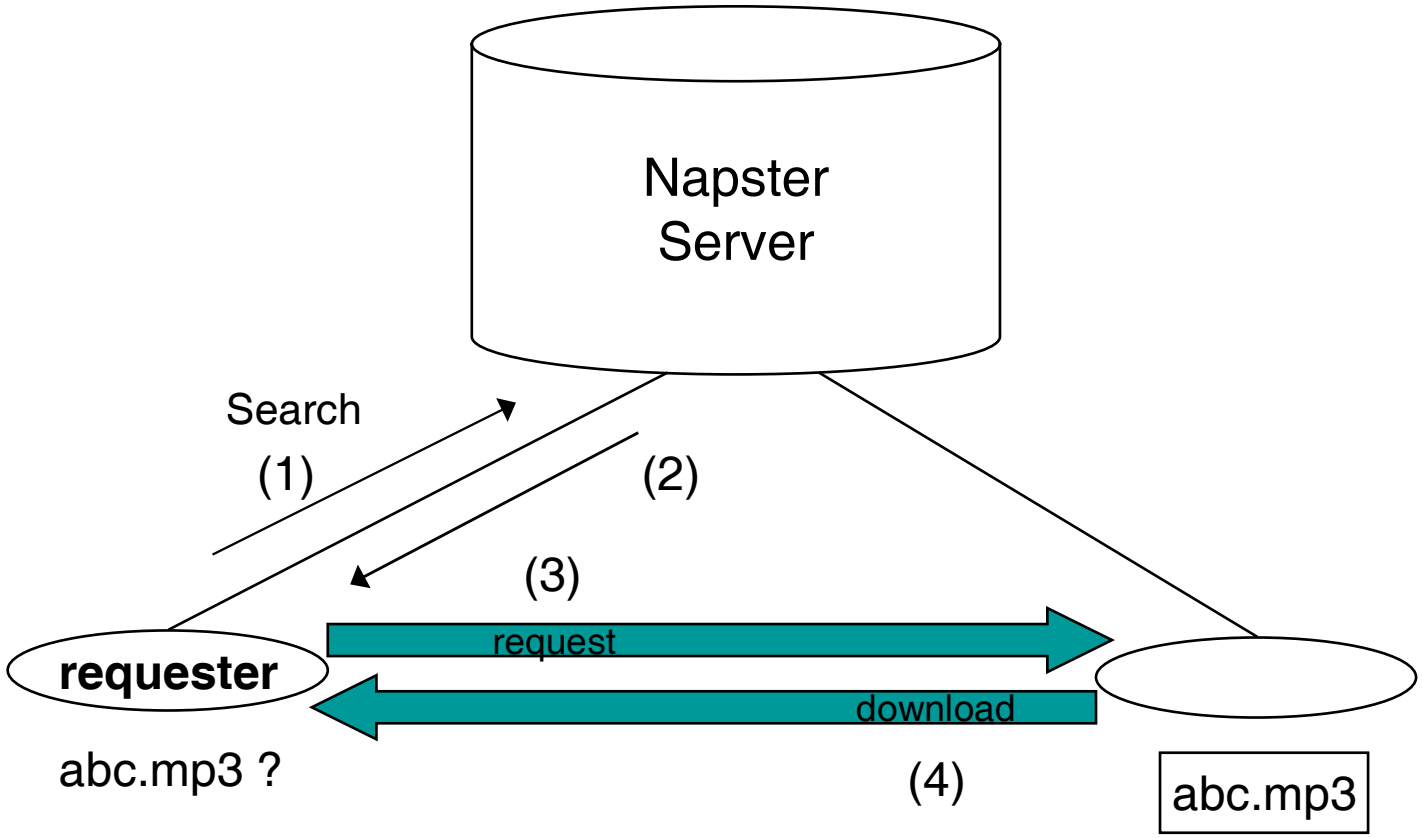
OUTLINE

- P2P computing systems
- Representative P2P systems
- P2P data management
- Incentive mechanisms
- Concluding remarks

NAPSTER

- First P2P file sharing system.
- Enables sharing of music files over the Internet.
- Uses a centralized directory to maintain the list of music files.
- Keyword based querying.
- Centralized index, distributed data.

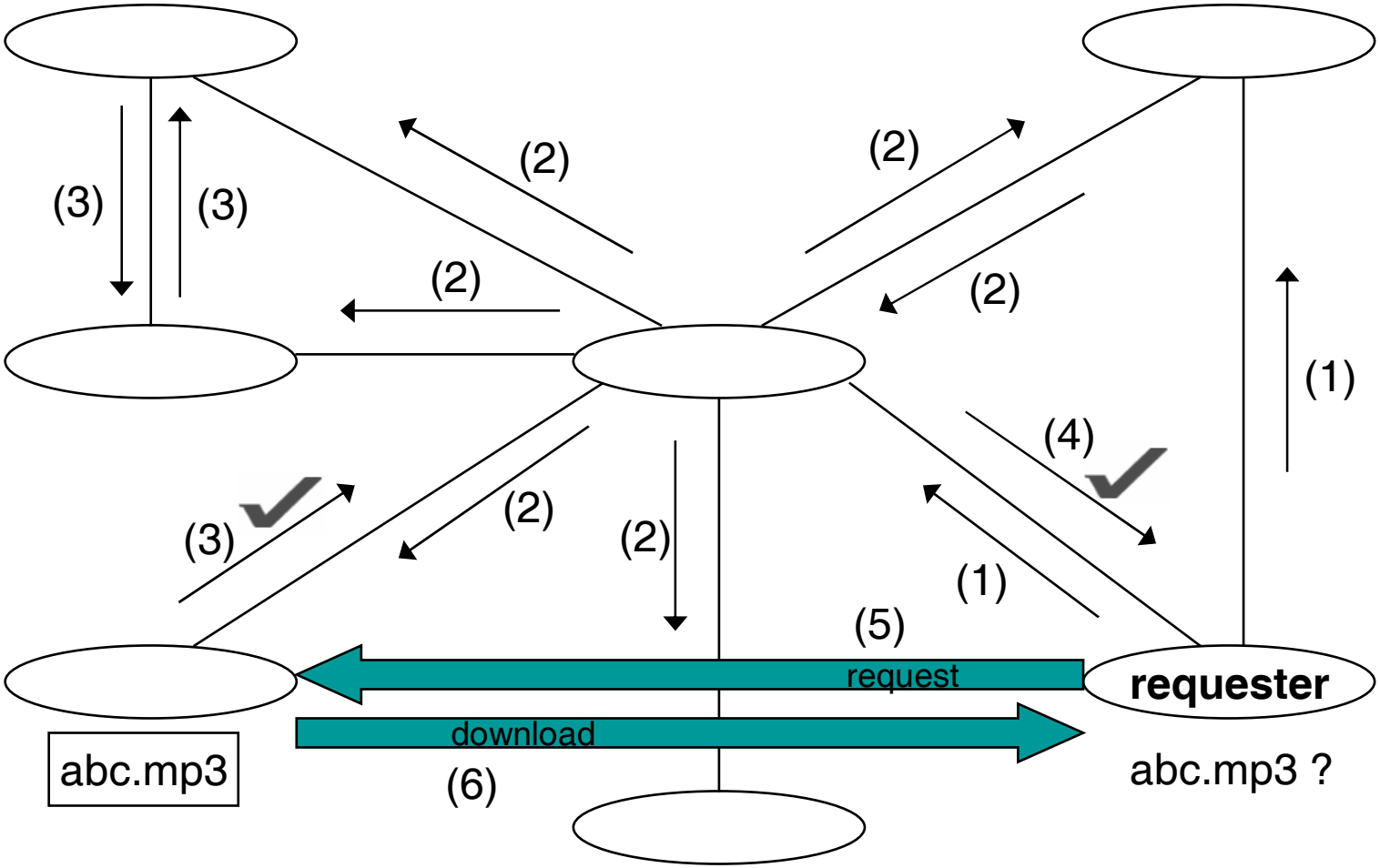
NAPSTER


Napster
Server
Search
(1)
(2)
(3)
request
requester
download
abc.mp3 ?
(4)
abc.mp3

GNUTELLA

- Purely decentralized architecture.
- A file sharing protocol.
- Users can search for and download files from the other users connected to the Internet.

GNUTELLA


(3)
(3)
(2)
(2)
(2)
(2)
(1)
(4)
(3)
(2)
(2)
(1)
(5)
request
requester
download
abc.mp3
(6)
abc.mp3 ?

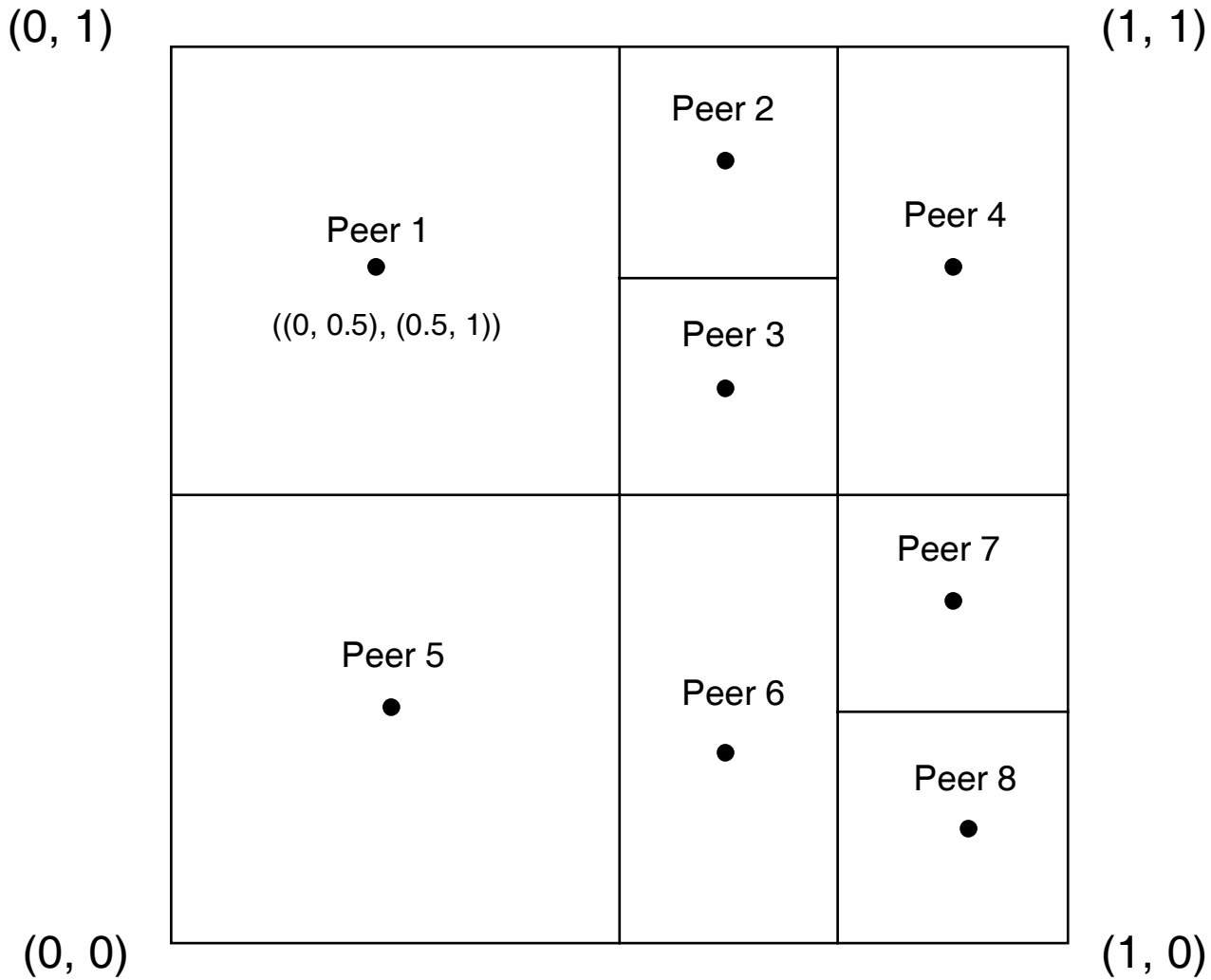
GNUTELLA

- Flooding to distribute messages.
- Time-to-live (TTL) to limit the spread of messages.
- Superpeers in more recent versions of Gnutella.
  - Shared files are indexed at superpeers.
  - Queries are initially directed to superpeers.

CAN (Content Addressable Network)

- A distributed hash table.
- Uses a d-dimensional coordinate space.
- Each peer is responsible for a portion of data space, called “zone”.
- A key is mapped onto a point in the coordinate space, and is stored at the peer whose zone contains the point’s coordinates.

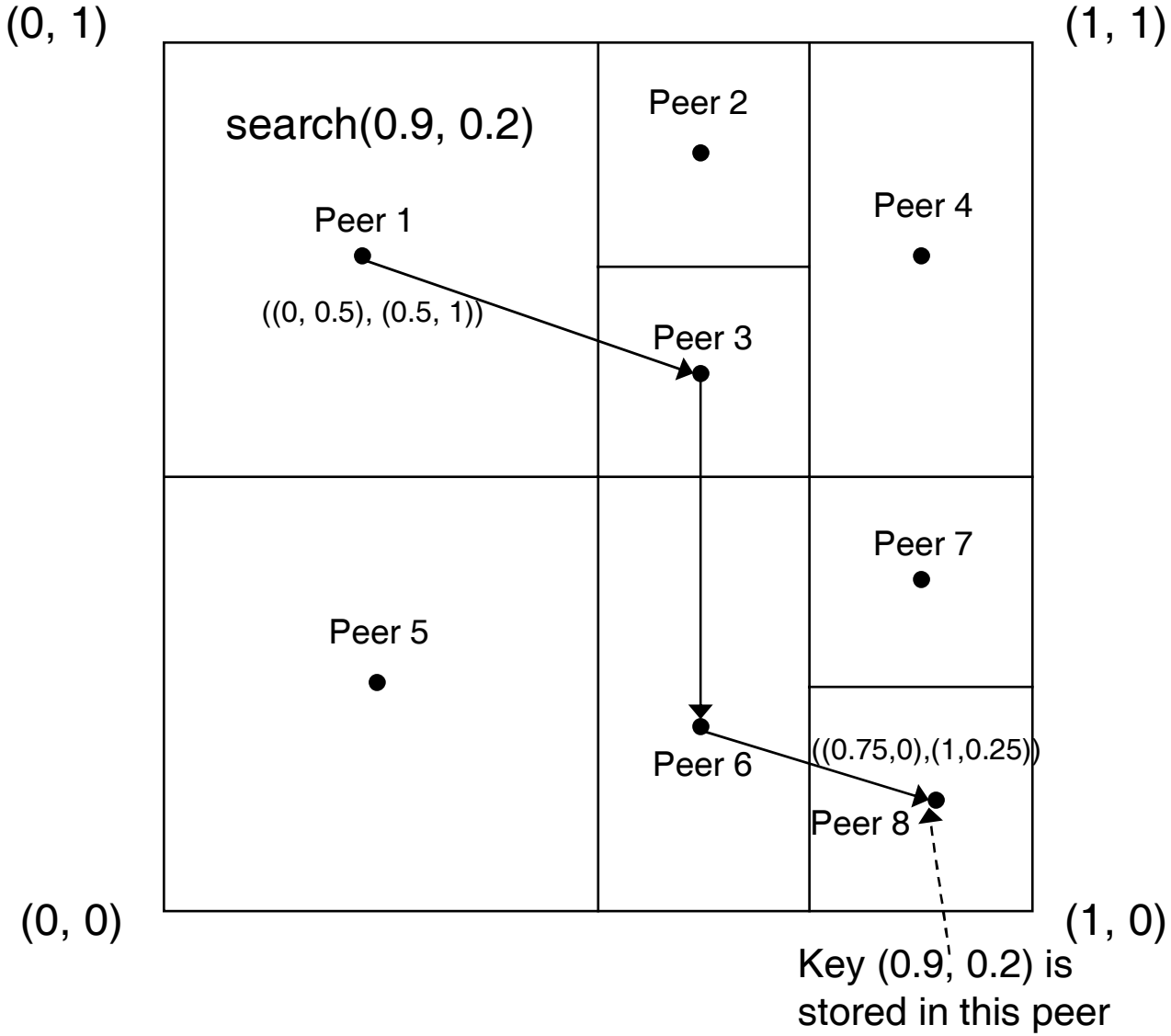
CAN (Content Addressable Network)



CAN (Content Addressable Network)

- Each peer maintains a routing table of its neighbors.
- Query messages are forwarded along a path to the source zone containing the key.

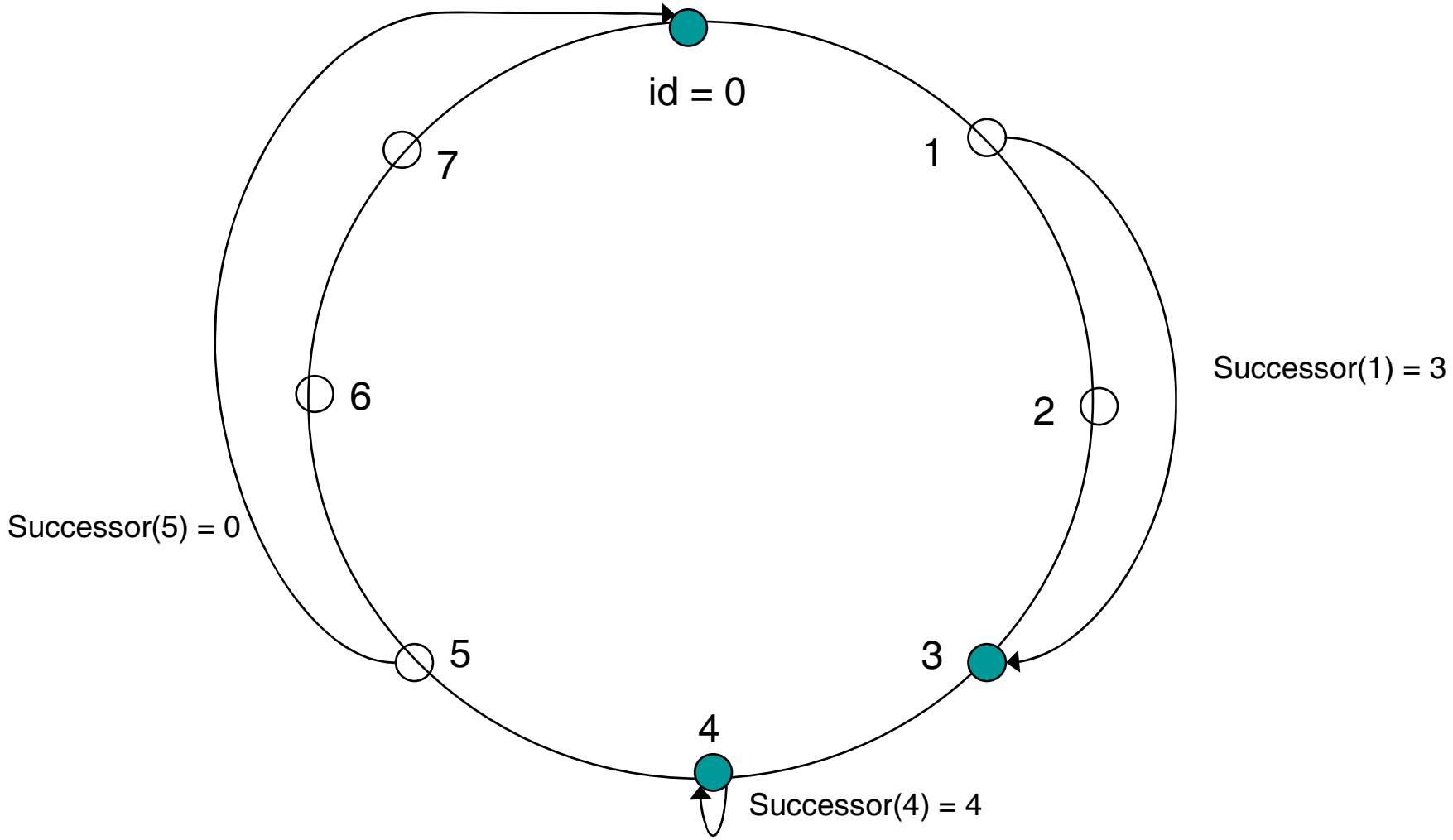
CAN (Content Addressable Network)



CHORD

- Peers form a ring called *identifier ring* or *Chord ring*.
- File keys are distributed over the same identifier space.
- *Successor*(*k*): The peer whose identifier is equal to or follows *k* in the identifier space. The data file with key *k* is assigned to Successor(*k*).

CHORD

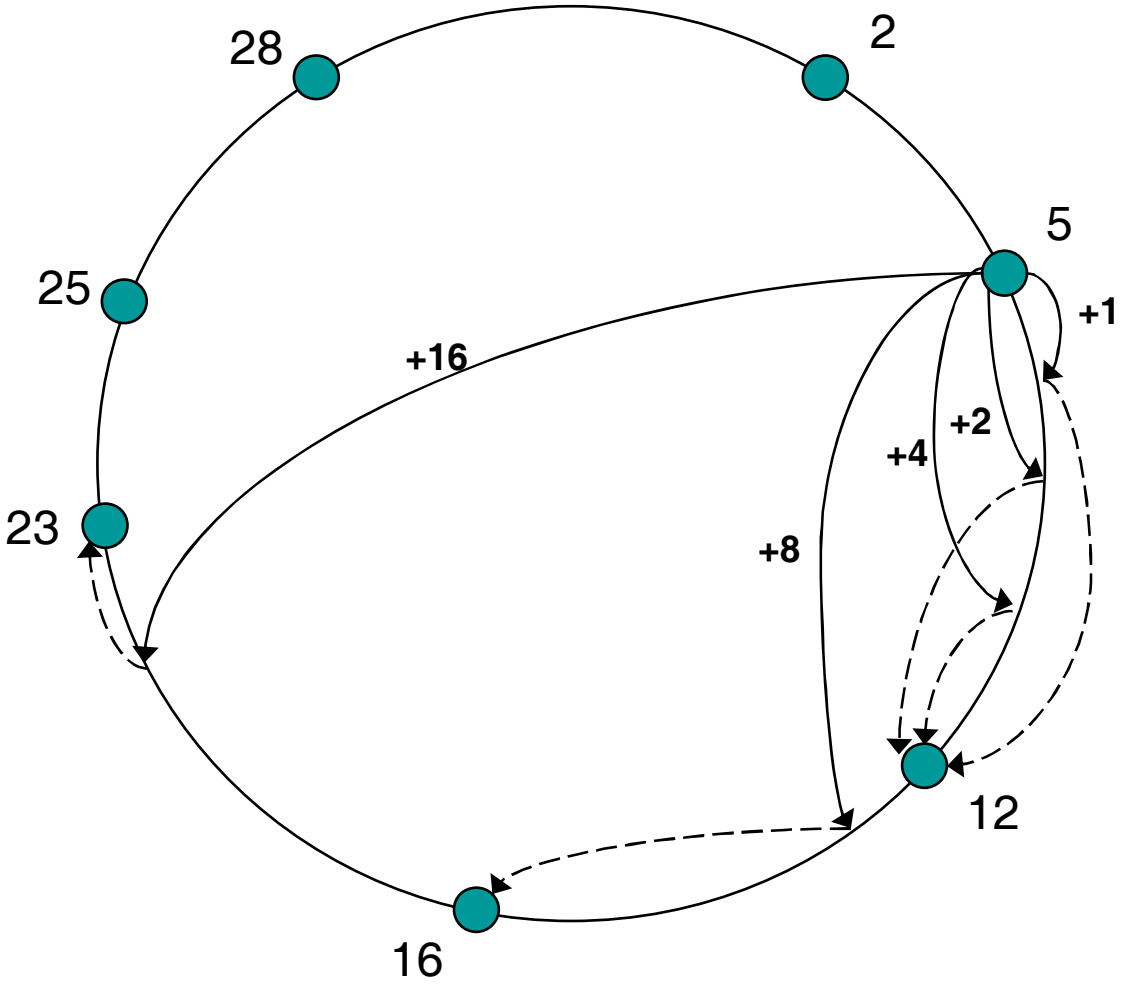


Keys 1, 4, 5 are located at peers 3, 4, 0 respectively.

CHORD

- A skiplist-like routing table, called *finger table*, is used by peers.
- Each peer keeps track of log(*N*) other peers in its finger table.

CHORD

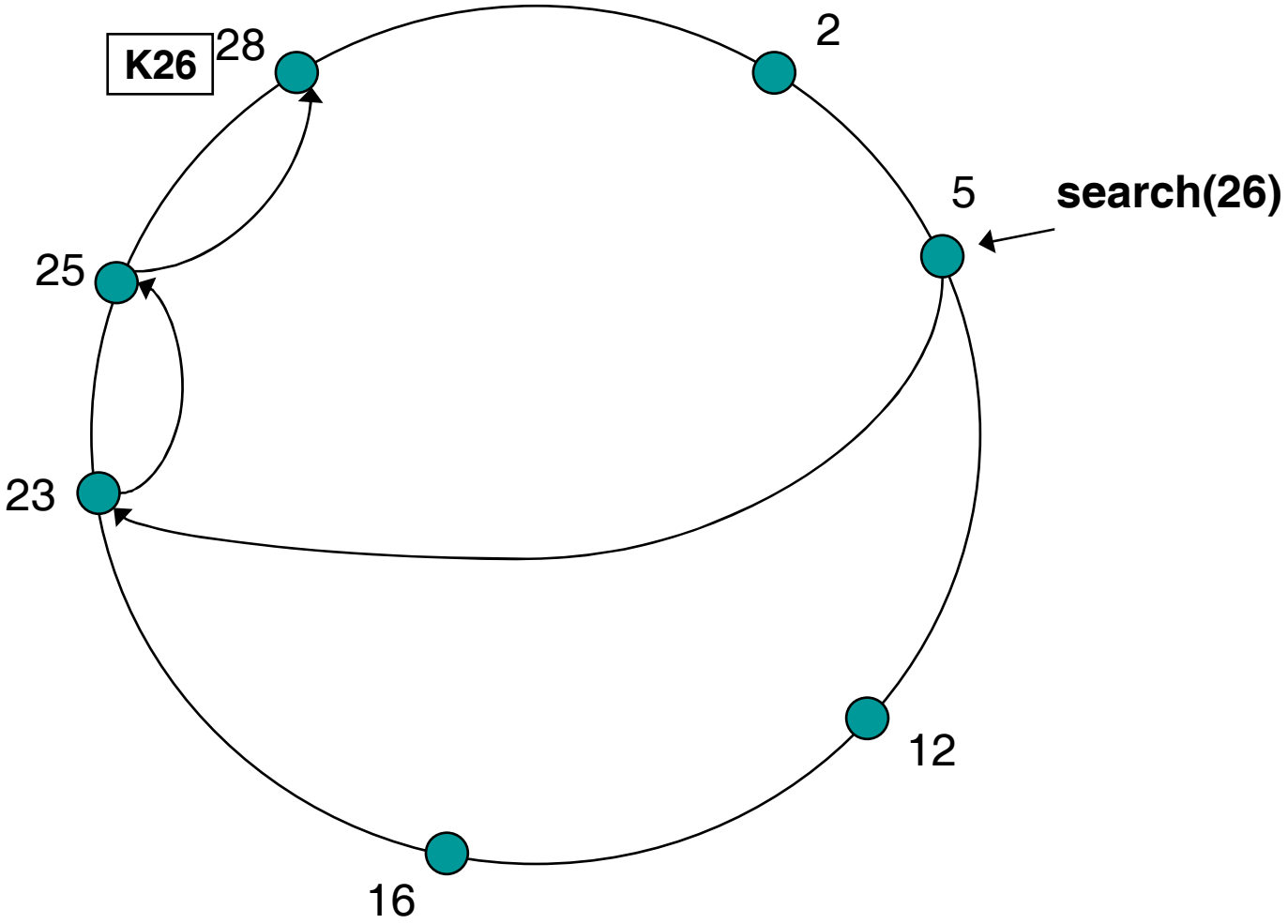


Finger Table of Peer 5	
5+1	12
5+2	12
5+4	12
5+8	16
5+16	23

CHORD

- Search operation is performed through finger tables.
- A peer forwards a query for key *k* to the peer in its finger table with the highest *id* not exceeding *k.*

CHORD



OUTLINE

- P2P computing systems
- Representative P2P systems
- P2P data management
- Incentive mechanisms
- Concluding remarks

OUTLINE

- P2P data management
  - Indexing
  - Data integration
  - Query processing
  - Replication
  - Clustering

INDEXING

- Traditional distributed systems use centralized or distributed indices.
- Indices in P2P systems:
  - must support frequent updates,
  - need to be highly scalable.
- Index types:
  - Local
  - Centralized
  - Distributed

LOCAL INDEX

- Peers index only their own data.
- Flooding is used for routing.
- Large volume of query traffic with no guarantee for matching.
- Scalability problem due to large network overhead.
- To improve scalability: fixed time-to-leave (TTL) rings, expanding rings, random walk, random k-walkers.

CENTRALIZED INDEX

- A single peer maintains references to data on all peers.
- Index is centralized but the data is distributed.
- The index server becomes a bottleneck and a single point of failure.
- Replicas of the centralized index may be maintained.
  - Higher reliability and scalability.
  - Huge number of updates is not handled efficiently.

DISTRIBUTED INDEX

- Index distribution can vary depending on
  - the structure of overlay network, and
  - the degree of centralization of the P2P system.

- Partially centralized P2P systems (e.g., Kazaa)
  - A superpeer maintains index information about a number of other peers it is responsible for.

DISTRIBUTED INDEX

- Structured P2P systems
  - Each peer maintains index for the data assigned to it.
  - *Distributed Hash Tables (DHT*s)
    - greedy routing
    - robustness
    - low diameter

OUTLINE

- P2P data management
  - Indexing
  - Data integration
  - Query processing
  - Replication
  - Clustering

DATA INTEGRATION

- Schema Mappings
  - Required as different names or formalisms are used to describe data.
  - Aims to provide uniform querying environment that hides heterogeneity.
  - Traditional approach is to use a global unified schema. Queries are specified interms of the global schema.

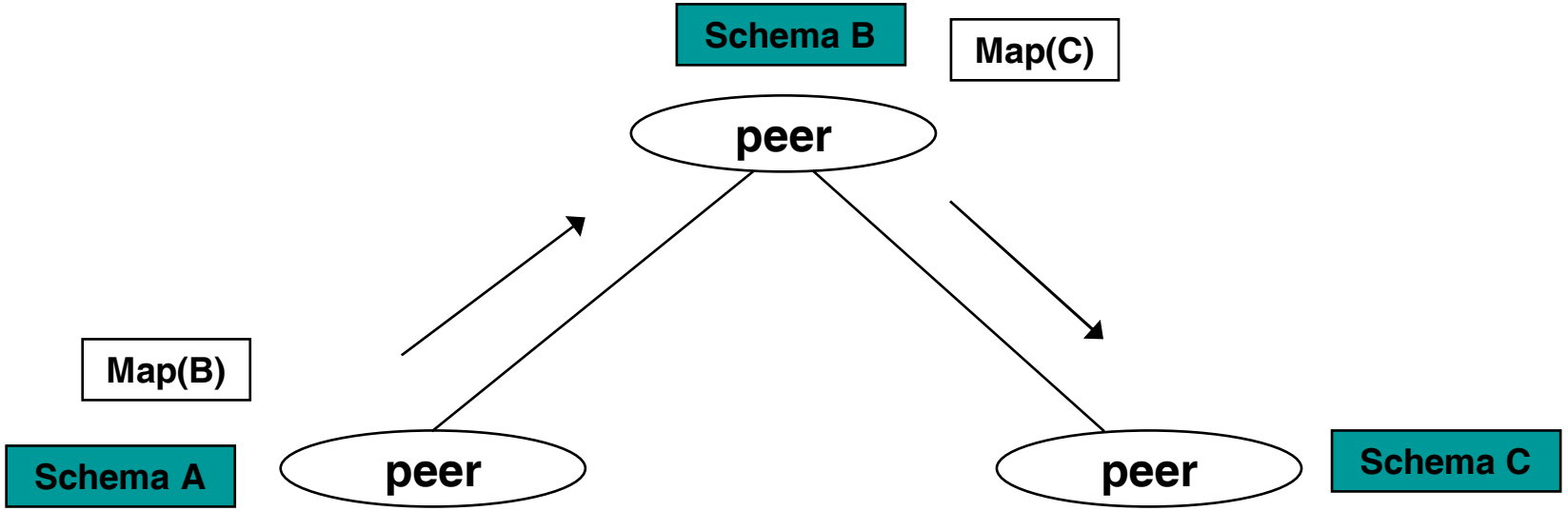
DATA INTEGRATION

- Schema Mappings (contd.)
  - A global unified schema is not applicable to P2P systems due to:
    - Autonomous nature of peers
    - Volatility of the system
    - Scalability of the system

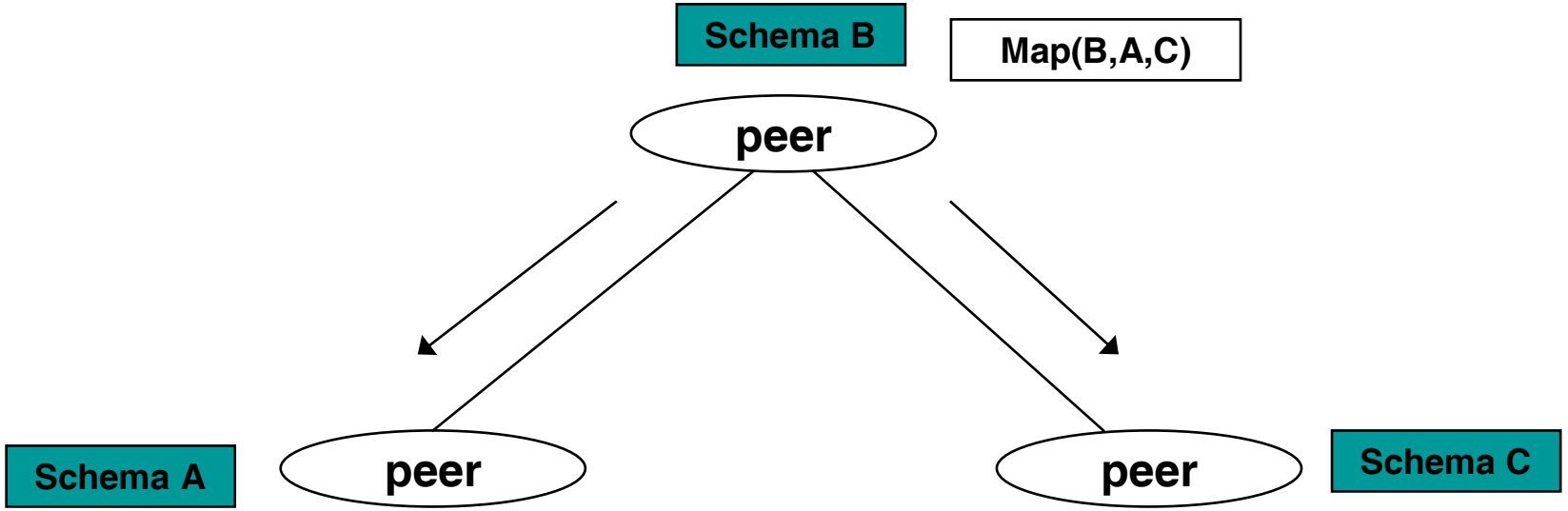
DATA INTEGRATION

- P2P Schema Mappings
  - Pair Mappings: only between pairs of peers.
  - Peer-Mediated Mappings: a generalization of pair mappings. (e.g., Piazza, PeerDB).
  - Super-Peer Mediated Mappings: at the super-peer level (e.g., Edutella).

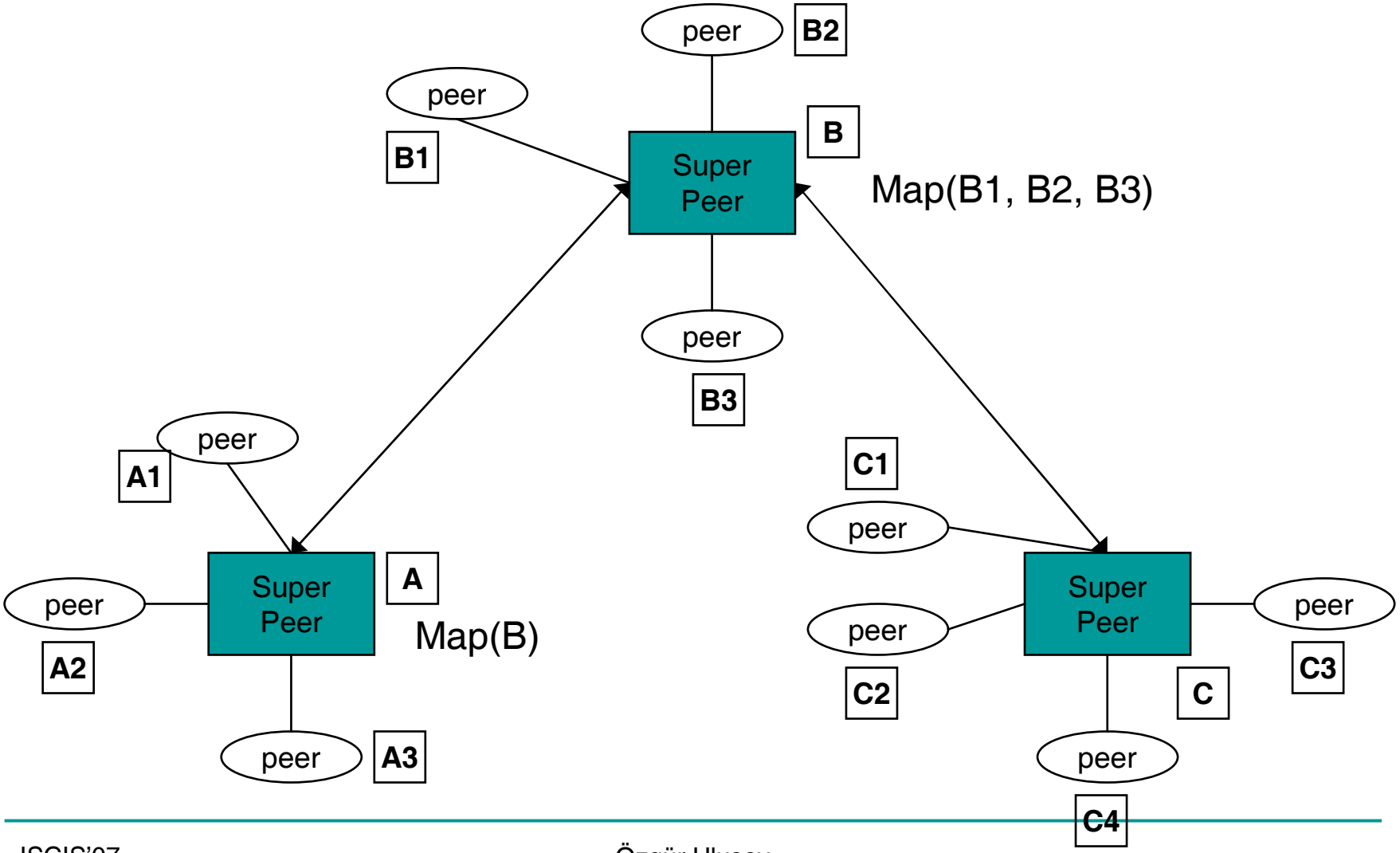
Pair Mappings



Peer-Mediated Mappings



SuperPeer-Mediated Mappings



OUTLINE

- P2P data management
  - Indexing
  - Data integration
  - Query processing
  - Replication
  - Clustering

QUERY PROCESSING

- Querying shared files
  - Queries are routed to peers to locate files.
- Querying structured data
  - More expressive queries are allowed.
- Current research in P2P systems:
  - key lookups in structured networks
  - keyword queries in unstructured networks
- Key lookups and keyword queries are not expressive enough.

QUERYING SHARED FILES

- Aim is to locate the source peers.
- Popular files are replicated.
- Unpopular files may not be found.
- Routing schemes:
  - Blind searches
  - Informed searches

QUERYING SHARED FILES

- Blind Search
  - Query is propagated arbitrarily to peers.
  - The query may not be satisfied.
  - No information about file locations is utilized.
  - Routing methods: Flooding, TTL rings, random walks.

QUERYING SHARED FILES

- Informed Search
  - Each peer maintains some form of routing table containing file placement information.
  - The chances for locating file increases as the hop count in the route increases.

QUERYING SHARED FILES

- Informed Search (contd.)
  - Search methods differ in the information collected about the peers and file placements.
    - *Query Routing Protocol* (*QRP*)

      Keywords describing the file contents that a peer offers are summarized in routing tables and exchanged with neighbors.
    - *Routing Indices*
    - *FreeNet's routing scheme*

QUERYING STRUCTURED DATA

- Support for complex query types is desirable
  (*range queries, multi-attribute queries, join queries, aggregation queries).*
- Multi-Attribute Addressable Network (MAAN)
  - Supports multi-attribute and range queries.
  - Built on CHORD structured P2P network.
  - A locality preserving hash function is used to map attribute values.

COMPLEX QUERIES

- Multi-Attribute Addressable Network (contd.)
  - A range query searching for files with attribute value in the range [*l, u*] is forwarded to peer *pl* where *l* has been hashed to.
  - All the files of *pl* that satisfy the range query are gathered in the result.
  - Next, the query is forwarded to the immediate successor of peer *pl* in the ring.
  - This process continues until the query reaches peer *pu* where *u* has been hashed to.

COMPLEX QUERIES

- Multi-Attribute Addressable Network (contd.)
  - In multi-attribute setting, data consists of a set of attributed and their respective values.
  - A multi-attribute query is treated as a combination of subqueries on each attribute dimension.
  - Subqueries are executed at appropriate peers and results are merged at query initiator.

COMPLEX QUERIES

- Supporting SQL in P2P Systems
  - PIER, PeerDB projects.
  - A great deal of additional research is needed to advance current work.

OUTLINE

- P2P data management
  - Indexing
  - Data integration
  - Query processing
  - Replication
  - Clustering

REPLICATION

- Aims to improve system performance, and increase data availability and reliability in case of peer failures.
- Classical issues of interest: what to replicate, granularity of replicas, where to place them, how to keep them consistent.
- Research challenges specific to P2P systems: high rates of peer joins and failures, lack of global knowledge, low online probability.

REPLICATION ISSUES IN P2P

- Data Consistency and Data Continuity
  - In Chord, *k* replicas are stored at *k* successors in the ring.
  - If the primary replica fails, the successor immediately takes over.
  - *Push update* to all the other peers storing the replicas, similar to a flooding scheme.
  - When become online, *pull update* from the most recent replica.

REPLICATION ISSUES IN P2P

- Number of Replicas
  - *Uniform replication*: same number of replicas are created for all data items.
  - *Proportional replication*: number of replicas created is proportional to the popularity of items.
  - *Square-root replication*: for any two data items, the ratio of replication is the square root of the ratio of their query rates.

REPLICATION ISSUES IN P2P

- Placing Replicas – Unstructured Networks
  - *Owner replication*: whenever a peer issues a successful query about a data item, a replica of the item is created at that peer. (e.g., Gnutella)
  - *Path replication*: copies of the requested data are stored at all peers along the path in the overlay network from the provider to the requestor. (e.g., FreeNet)

REPLICATION ISSUES IN P2P

- Placing Replicas – Structured Networks
  - Issues addressed by replication: load balancing, data availability.
  - Some peers may be assigned with many popular data items by distributed hashing, and thus may become hot spots.
  - CAN uses multiple hash functions to assign each item to more than one peer.

OUTLINE

- P2P data management
  - Indexing
  - Data integration
  - Query processing
  - Replication
  - Clustering

CLUSTERING

- *Clustering data*: similar data are placed in neighboring peers.
- *Clustering peers*: peers with similar interests or similar data items are placed nearby in the overlay network.

CLUSTERING DATA

- *Order preserving hash functions* can be used to store similar documents at the same or neighboring peers.
- *Content-based clustering* is achieved by using a semantic vector describing the contents of files, as the input of the hash function.

CLUSTERING PEERS

- *Intra-cluster organization*: specification of how the peers within a cluster are connected.
- *Inter-cluster organization*: specification of how the clusters are connected.
- *Two-step routing*: identification of appropriate cluster, and then routing the query within that cluster.

CLUSTERING PEERS

- *Peer communities*
  - Membership depends on the relationship between peers that share common interests.
  - Clustering is implemented through sets of *attributes* that the peers choose.

CLUSTERING PEERS

- *Semantic overlay clustering*
  - Based on superpeer networks.
  - Clusters of peers with similar semantic profiles are formed.
  - Each superpeer acts as *cluster representative*, which is in charge of the management of the cluster for query processing.

CLUSTERING

- P2P Challenges of Clustering
  - Autonomy of peers
    - Data clustering violates storage autonomy.
  - Dynamic nature of connections and content
    - Clustering method must be dynamic and incremental.
  - Lack of global knowledge
    - Clustering algorithms assume global knowledge of data.

OUTLINE

- P2P computing systems
- Representative P2P systems
- P2P data management
- Incentive mechanisms
- Concluding remarks

FREE RIDING

- *Free riding*: exploiting P2P network resources without contributing to the network at an acceptable level.
- Effects on P2P networks
  - A small number of peers serves a large number of requests:
    - scalability problem
    - degeneration to client-server like paradigm
  - Renewal or presentation of new content may decrease in time:
    - limited grow in the number of shared files

FREE RIDING

- Effects on P2P networks (contd.)
  - Quality of search process may degrade:
    - less satisfaction in query results
    - decrease in peer population
  - Network traffic increases:
    - degradation of P2P services
    - delay, congestion and loss of messages

FREE RIDING STATISTICS

- Gnutella
  - 85% of peers do not share any files at all.
  - Top 1% of the sharing peers provide 50% of all query responses, while the top 25% provide 98%.
- Napster
  - About 20-40% of the Napster peers share little or no files.
  - 30% of the users that reported their bandwidth as 64 Kbps or less, but they actually had a significantly higher bandwidth.
- Similar findings were reported for KaZaA, Maze, eDonkey networks.

INCENTIVE MECHANISMS

- Methods to encourage peer cooperation
  - *Micropayment-based*
  - *Reciprocity-based*
  - *Reputation-based*

INCENTIVE MECHANISMS

- *Micropayment-based Approaches*
  - Peers are required to pay for the services they get or resources they consume.
  - Incentive: peers are charged for every download and paid for every upload.
  - A trusted third-party is required for tracking accounts, distributing virtual currency, and providing security.

INCENTIVE MECHANISMS

- *Reciprocity-Based Approaches*
  - Each peer decides how to serve any other peer based on the direct service exchange with this peer in the past.
  - Example: Tit-for-Tat mechanism of *BitTorrent*
    - A peer uploads to the peers that have given him the best downloading rate. The other peers are disallowed from downloading.

INCENTIVE MECHANISMS

- *Reputation-Based Approaches*
  - A reputation rating is produced for the peers.
    Peers with low reputation are excluded from the network.
  - Reputation information is locally generated, and it is spread throughout the network.
  - Implementation issues: security and availability of reputation information, identity management, communication overhead.

OUTLINE

- P2P computing systems
- Representative P2P systems
- P2P data management
- Incentive mechanisms
- Concluding remarks

CONCLUDING REMARKS

- Addressed in this talk:
  - Key research issues relevant to P2P data management
  - Associated challenges
  - **Open research problems and directions**:
    - Design of effective incentive mechanisms and reputation systems.
    - Extension of the key-value lookup and building application flexibility into Distributed Hash Tables (DHTs).
    - Creation of semantic mappings to handle heterogeneity in shared data sets.
    - Adaptation of more expressive queries.
    - Methods to maintain replica consistency in the face of P2P specific challenges.

THANK YOU

REFERENCES

- R. Blanco et al. *A Survey of Data Management in Peer-to-Peer Systems*, Technical Report CS-2006-18, David R. Cheriton School of Computer Science, University of Waterloo, Waterloo, Ontario, Canada, 2006.
- M. Cai et al. “Maan: A Multi-Attribute Addressable Network for Grid Information Services”, *IEEE International Workshop on Grid Computing (GRID*), pp.184-191, 2003.
- E. Cohen, S. Shenker, “Replication Strategies in Unstructured Peer-to-Peer Networks”, *ACM Conference on Applications, Technologies, Architectures, and Protocols for Computer Communication (SIGCOMM),* pp.177-190, 2002.
- B. Cohen, “ Incentives Build Robustness in Bittorrent”, *Workshop on Economics of Peer-to-Peer Systems*, 2003.
- F. Dabek et al. “Building Peer-to-Peer Systems with Chord, a Distributed Lookup Service”, *IEEE Workshop on Hot Topics in Operating Systems (HoTOS*), pp.81-86, 2001.

REFERENCES

- A. Datta et al. “Updates in Highly Unreliable, Replicated Peer-to-Peer Systems”, *International Conference on Distributed Computing Systems (ICDCS),* pp.76-85, 2003.
- D. Hughes et al. “Free Riding on Gnutella Revisited: the Bell Tolls?”, *IEEE Distributed Systems Online*, vol.6, no.6, 2005.
- M. Karakaya, I. Korpeoglu, Ö. Ulusoy, “A Distributed and Measurement-Based Framework Against Free Riding in Peer-to-Peer Networks”, *IEEE International Conference on Peer-to-Peer Computing*, 2004.
- M. Karakaya, I. Korpeoglu, Ö. Ulusoy, “A Connection Management Protocol for Promoting Cooperation in Peer-to-Peer Networks”, *Computer Communications*, to appear, 2007.
- M. Khambatti et al. “Structuring Peer-to-Peer Networks Using Interest-Based Communities”. *International Workshop on Databases, Information Systems and Peer-to-Peer Computing (DBISP2P),* pp.48-63, 2003.

REFERENCES

- G. Koloniari, E. Pitoura, “Peer-to-Peer Management of XML Data: Issues and Research Challenges”, *ACM SIGMOD Record*, vol.34, no.2, pp.6-17, 2005.
- G. Koloniari et al. *Overlay Networks and Query Processing: A Survey*, Technical Report TR2006-08, Computer Science Department, University of Ioannina, 2006.
- W. S. Ng et al. “Peerdb: A P2P-Based System for Distributed Data Sharing”, *International Conference on Data Engineering (ICDE),* pp.633-644, 2003.
- S. Ratnasamy et al. “A Scalable Content-Addressable Network”, *ACM Conference on Applications, Technologies, Architectures, and Protocols for Computer Communication (SIGCOMM),* pp.161-172, 2001.
- J. Risson, T. Moors, “Survey of Research towards Robust Peer-to-Peer Networks: Search Methods”, *Computer Networks*, vol.50, no.17, pp.3485-3521, 2006.
- C. Rohrs, “Query Routing for the Gnutella Network”, Unpublished Work. Available at http://rfc-gnutella.sourceforge.net/src/qrp.html, 2002.

REFERENCES

- S. Saroiu et al. “Measuring and Analyzing the Characteristics of Napster and Gnutella Hosts”, *Multimedia Systems*, vol.9, no.2, pp.170-184, 2003.
- I. Stoica et al. “Chord: A Scalable Peer-to-Peer Lookup Service for Internet Applications”, *ACM Conference on Applications, Technologies, Architectures, and Protocols for Computer Communication (SIGCOMM),* 2001.
- I. Tatarinov et al. “The Piazza Peer Data Management Project”, *ACM SIGMOD Record*, vol.32, no.3, 2003.
- S. Androutsellis-Theotokis, D.Spinellis, “A Survey of Peer-to-Peer Content Distribution Technologies”, *ACM Computing Surveys,* vol.36, no.4, pp.335-371, 2004.
- P. Valduriez, E. Pacitti, “Data Management in Large-Scale P2P Systems”, *High Performance Computing for Computational Science (VECPAR),* pp.104-118, 2004.