Peer to Peer Applications: KaZaA

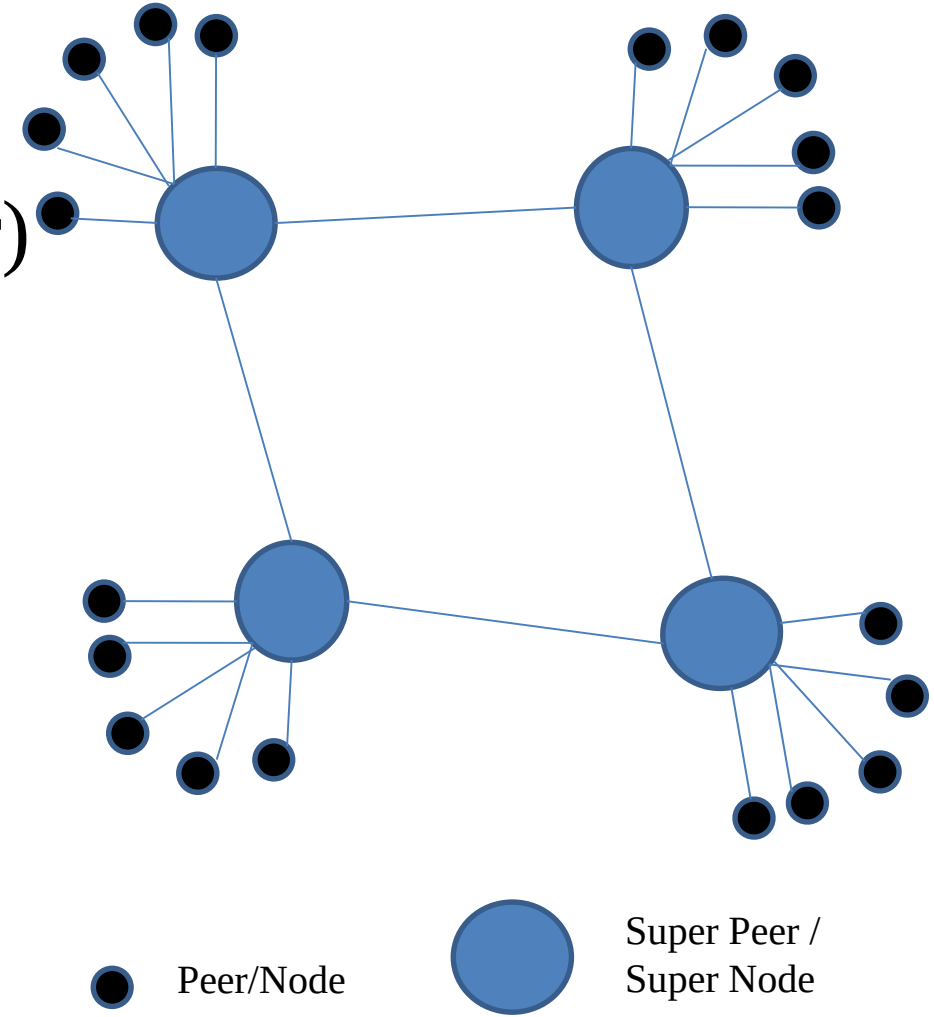
Kameswari Chebrolu

All the figures used as part of the slides are either self created or from the public domain with either 'creative commons' or 'public domain dedication' licensing. The public sites from which some of the figures have been picked include: http://commons.wikimedia.org (Wikipedia, Wikimedia and workbooks); http://www.sxc.hu and http://www.pixabay.com

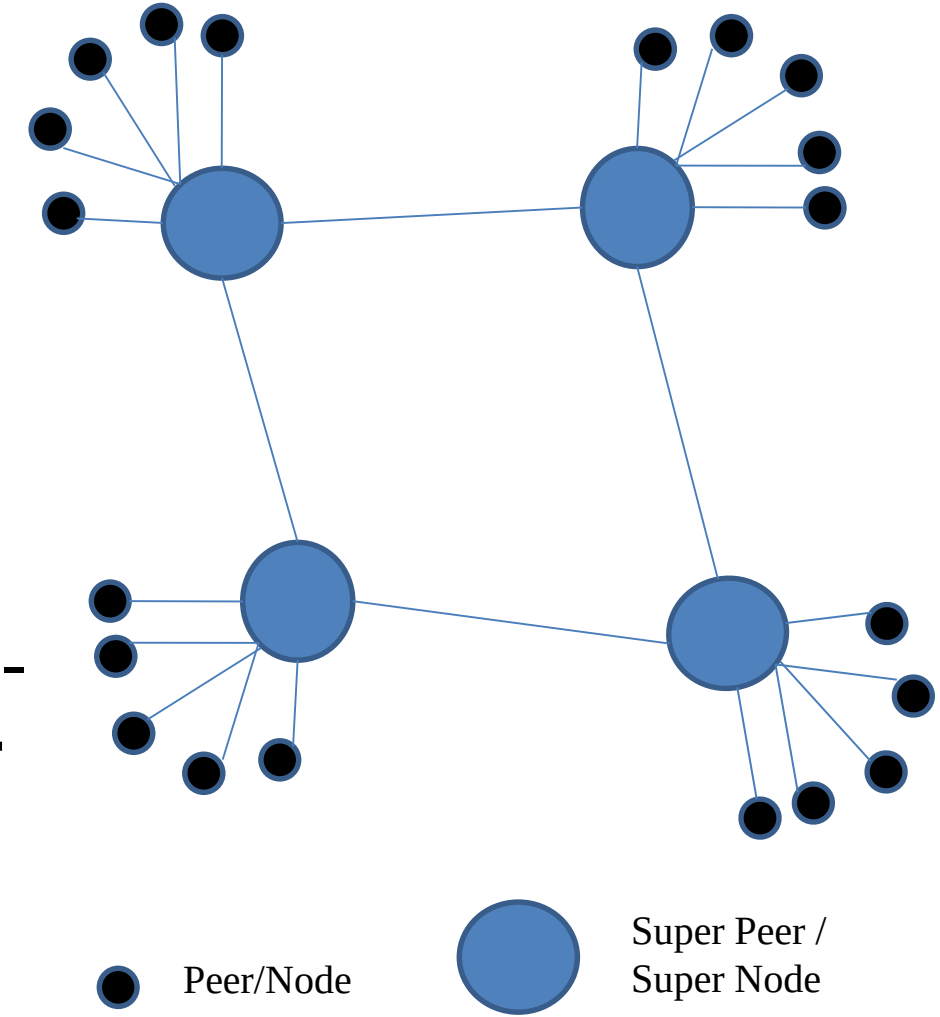
KaZaA

- Cross between Napster and Gnutella
  - Napster (centralized): efficient search but single point of failure
  - Gnutella (distributed): inefficient search based on flooding but robust against failure
    - All peers are treated equal → peers with limited resources can deteriorate overall performance
- Basic idea: Leverage heterogeneity of peers (assign more responsibility to peers with better resources –> super nodes/peers) and impose a hierarchy

- Super node acts as a central server to a subset of clients (like in Napster)
  - Search is efficient
  - A node may become super-node at some point based on its capabilities and how long its been on
  - A supernode serves about 100-150 nodes



- Super-nodes form a distributed network (like in Gnutella)
  - Many super-peers share load and provide robustness
  - A supernode has about 30-50 connections with other supernodes



Node Join

- Some supernodes hardcoded in downloaded software
- New peer contacts (hardcoded) supernodes one by one till it finds an operational supernode
- New peer obtains a more up-to-date list (200 entries) from the (hardcoded) supernode, contacts supernodes (in list) and connects to one who replies to it
- Supernodes typically are mostly ON, but in case it goes down, node chooses another supernode in list

Publish

- Node informs supernode of the list of files it wishes to share
- For each file, it uploads the following metadata
  - File name
  - File size
  - File hash (helps determine identical copies and permits parallel downloads)
  - File descriptors: used during keyword search

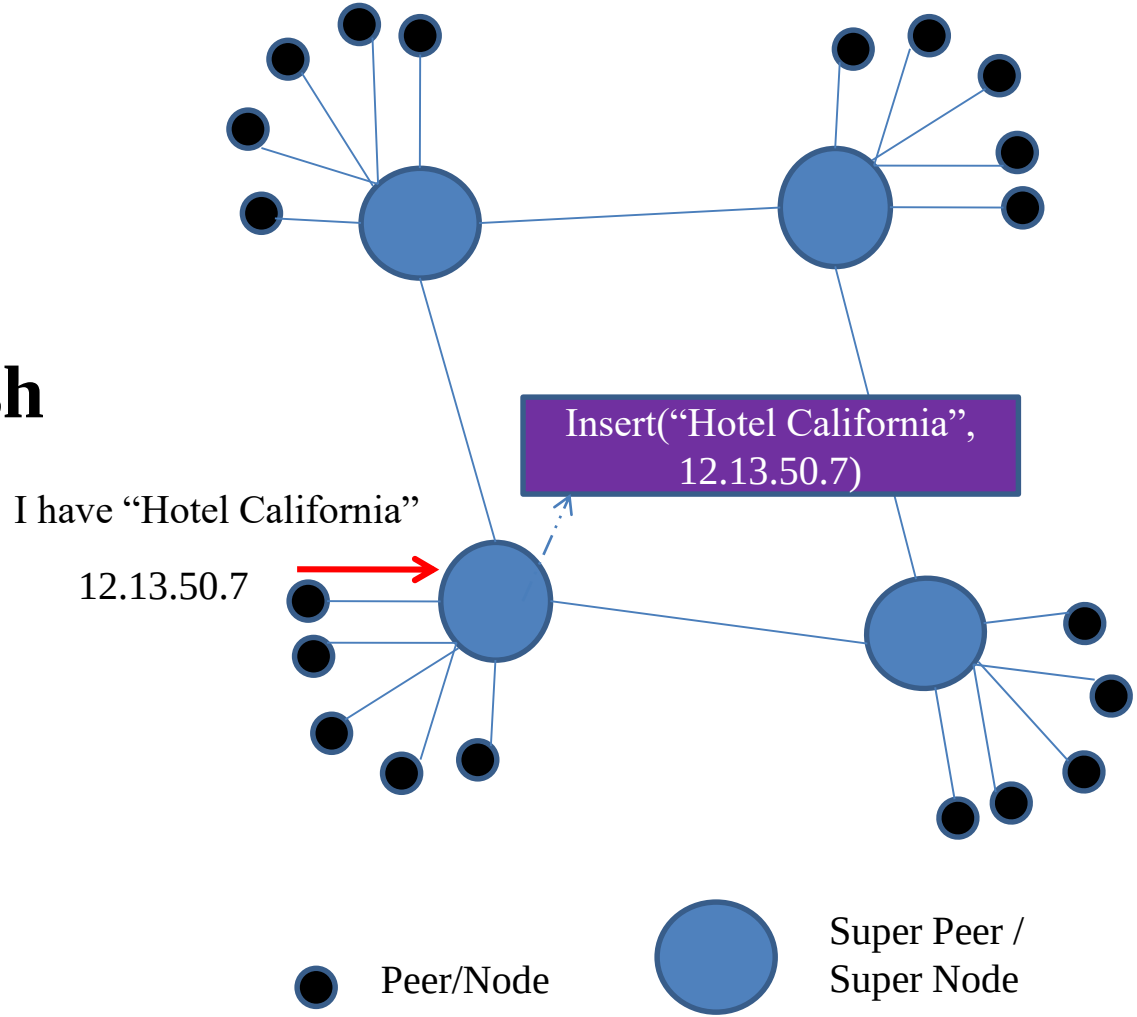
Search

- Node first queries the supernode; if x matches found done.
- Else, supernode forwards query to subset of supernodes (limited scope flooding in incremental steps) till a total of x matches found
- Supernode responds to node with matches
  - Each match returns metadata and IP address of peer with file

Fetch

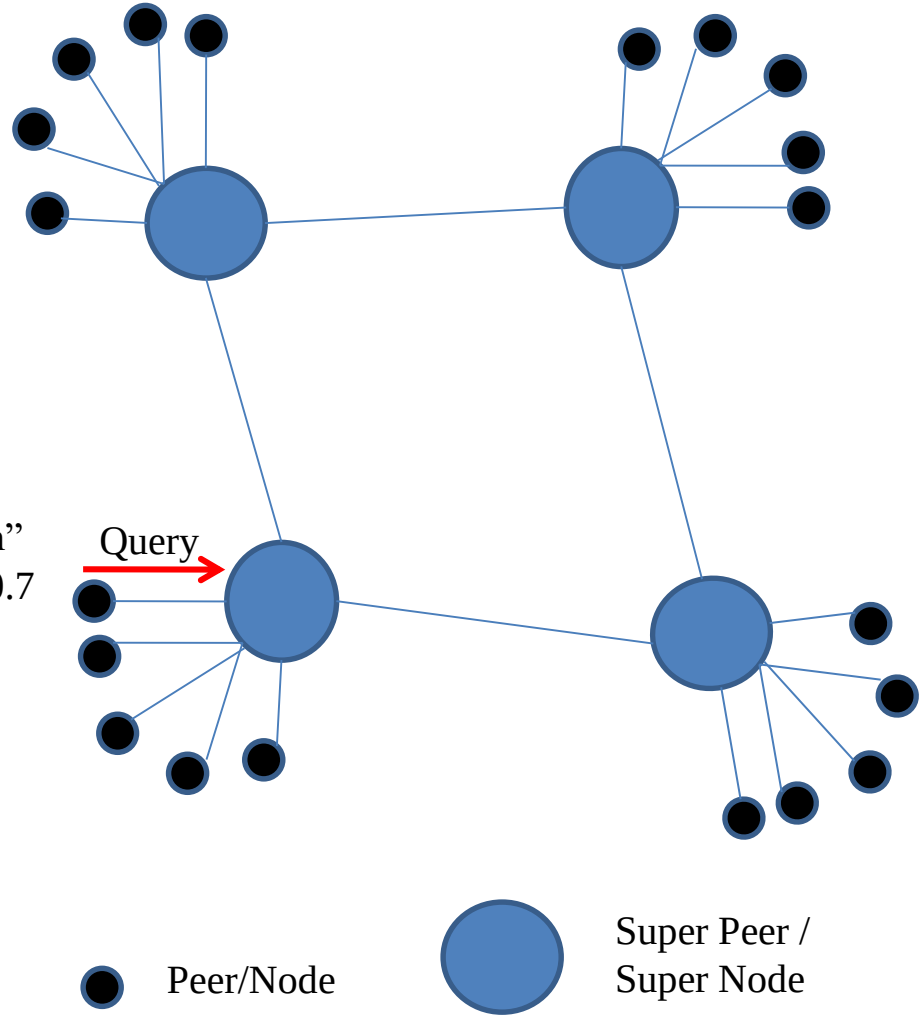
- Get file directly from peer(s)
- Parallel downloads permitted via HTTP byte-range header
  - Identical copies determined by file hash
  - HTTP byte-range header used to request different portions of the file from different nodes

Publish


Insert(“Hotel California”, 12.13.50.7)
I have “Hotel California”
12.13.50.7
Peer/Node
Super Peer / Super Node

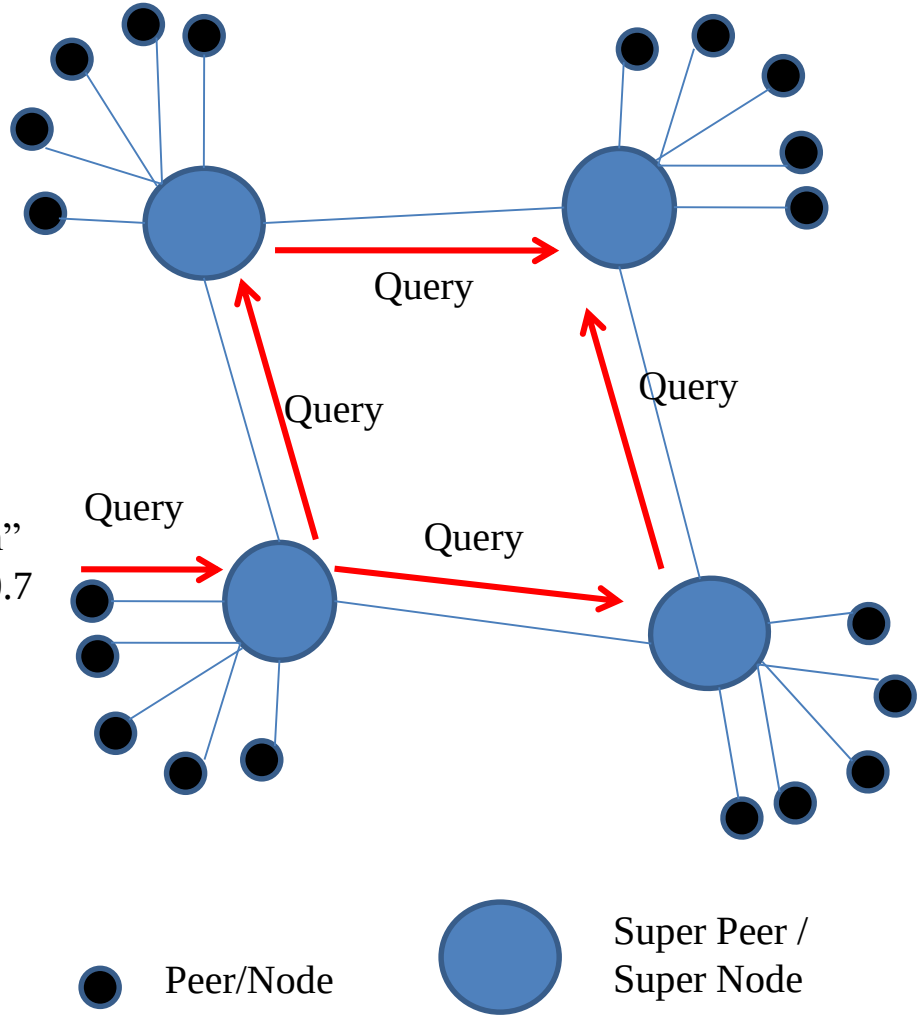
Search

I want “Stairway to Heaven”
12.13.50.7



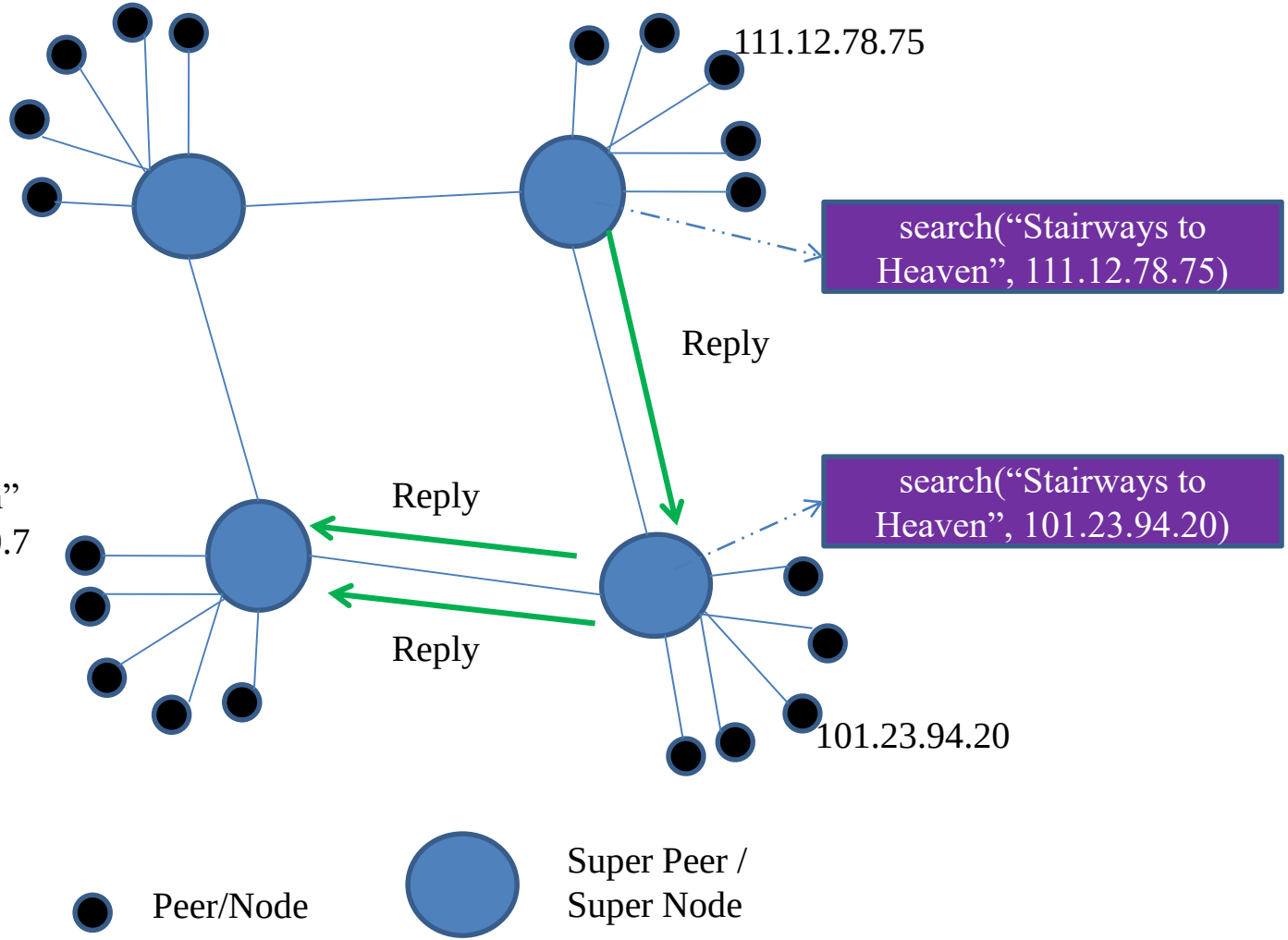
Search

I want “Stairway to Heaven”
12.13.50.7



Search

I want “Stairway to Heaven”
12.13.50.7



Discussion

Pros:

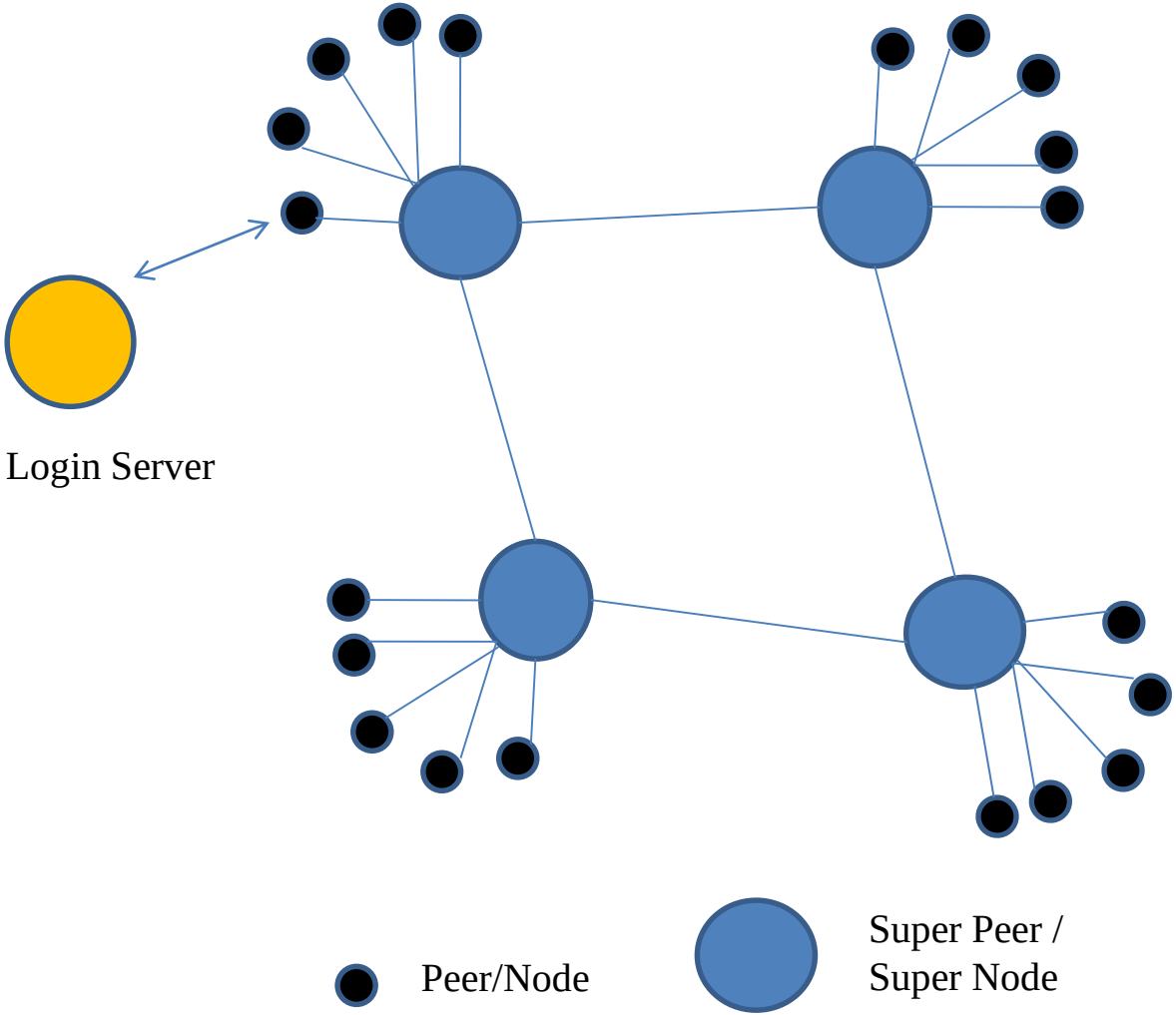
- Improves scalability and search efficiency by leveraging node heterogeneity

Cons:

- No guarantees on search time
- No mechanism to tackle freeloading

Skype

- Provides video/voice chat/calls from computers/mobile devices to other devices including telephones via Internet
- Same developers as KaZaA; similar architecture as KaZaA
  - Supports a few additional servers: login (for authentication), skypeout, skypein (for interfacing with PSTN)
- Provides excellent NAT and firewall traversal mechanisms


Login Server
Peer/Node
Super Peer /
Super Node

Summary

- KaZaA a cross between Napster and Gnutella
- Leverages heterogeneity of nodes and imposes a hierarchy in the overlay network
  - Supernodes form a distributed network and act as a central server to a subset of clients/peers
  - Improves scalability and searchability
- Popular IP telephony service Skype is P2P based whose working is very similar to KaZaA