



The diagram consists of two horizontal bars stacked vertically. Each bar has a dark gray vertical bar on its left side and a thin green vertical line next to it. The top bar is labeled 'P2P' in green text on the right side. The bottom bar is labeled 'SAD' in blue text on the right side.

P2P

SAD



Peer to Peer (P2P) Networks

- ▶ Favored structure
 - ▶ All nodes are born equal (peers)
 - ▶ No node is a “server”
 - ▶ Self organization
 - ▶ In some structures, nodes specialize
- ▶ ULTIMATE GOAL
 - ▶ Use resources of all network nodes efficiently
 - ▶ Storage, Processing power, etc...
 - ▶ Note that this goal does not necessarily impose structure...



Peer to Peer (P2P) Networks

- ▶ But also
 - ▶ Scale
 - ▶ Resilience
- ▶ Decentralized structure?
 - ▶ Various approaches



P2P: Availability of resources

▶ Centralized elements

▶ Make resilience suffer

- ▶ Failure of servers
- ▶ Failure of disks

▶ Approaches?

- ▶ Element replication,
 - Server replication forming clusters
 - Disk replication (RAID arrays)
 - Mixture

▶ ...However

- ▶ Even clustering approaches do not remedy this fully
 - Network failures toward the cluster...
 - Need network redundancy too

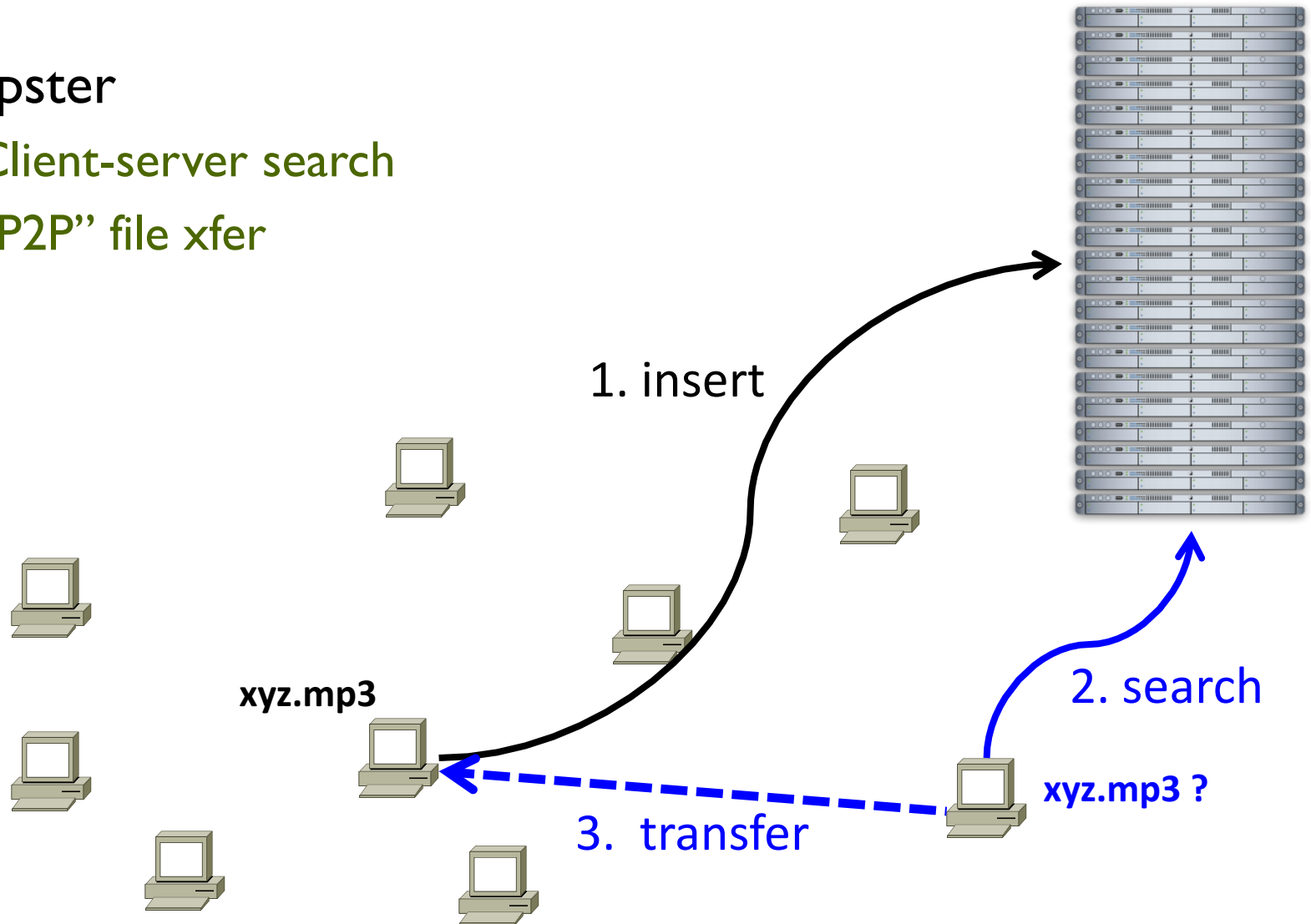
- ▶ Decentralized approach
 - ▶ However, for resilience
 - ▶ Still needs replication
 - failures may occur
 - ▶ Access to a resource does not depend on any single failure
 - ▶ But it may depend on a specific failure
 - Unless judicious replication is used
 - Replication on agents with failures unlikely correlated
 - ▶ Access to different resources from different agents do not interfere with each other

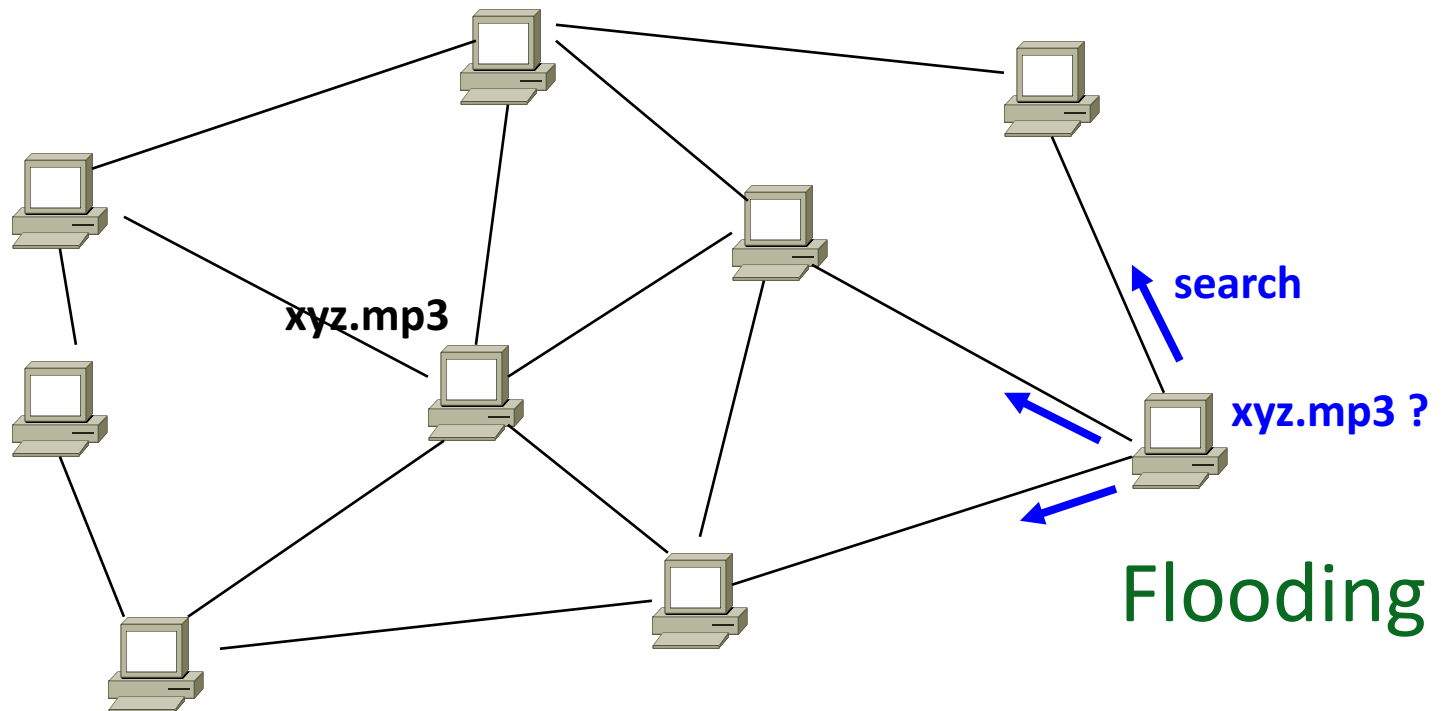


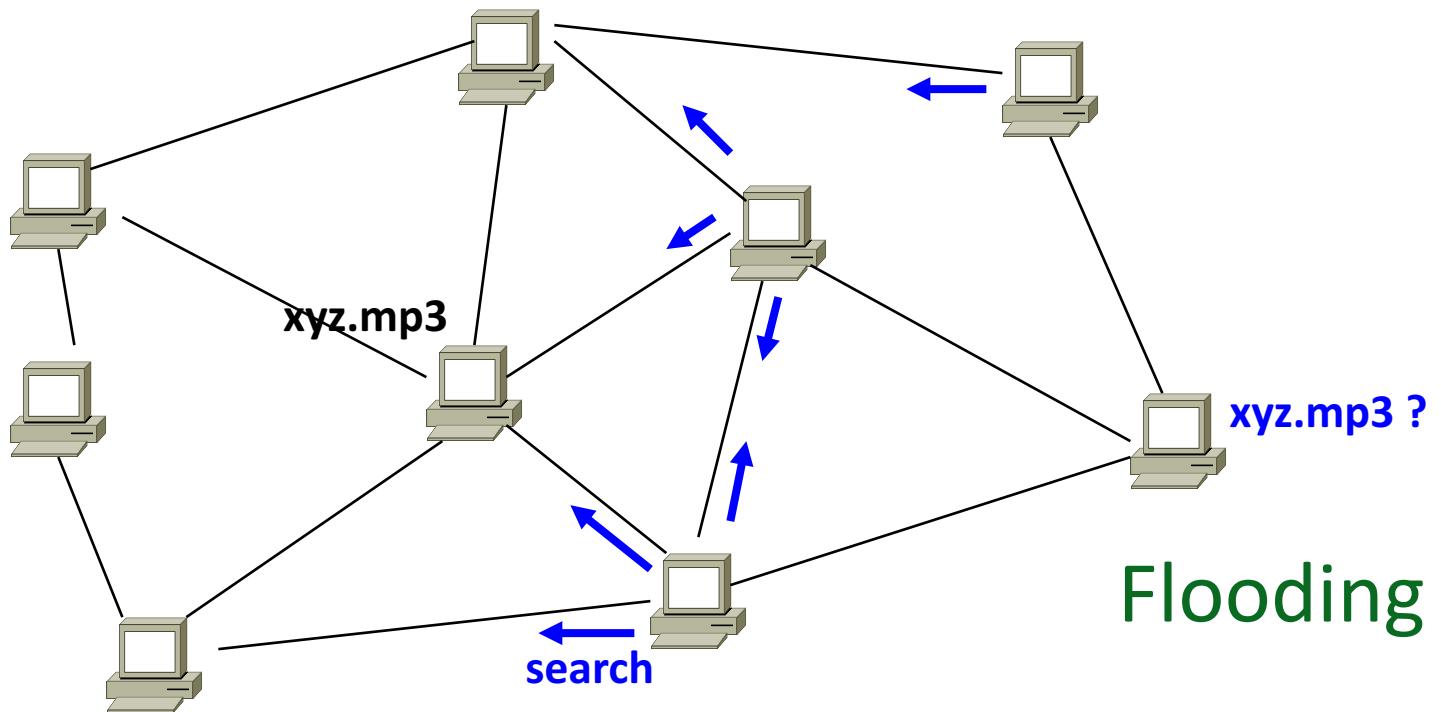
Generic P2P Classification

- ▶ **Centralized**
 - ▶ E.g. SETI@HOME
 - ▶ Long stretch to call this a P2P
- ▶ **Mediated/Hybrid**
 - ▶ Centralized location
 - ▶ Decentralized access
- ▶ **Decentralized**
 - ▶ True P2P
 - ▶ Usually via overlay networks
 - ▶ Structured/Unstructured
 - ▶ Decentralized location & access

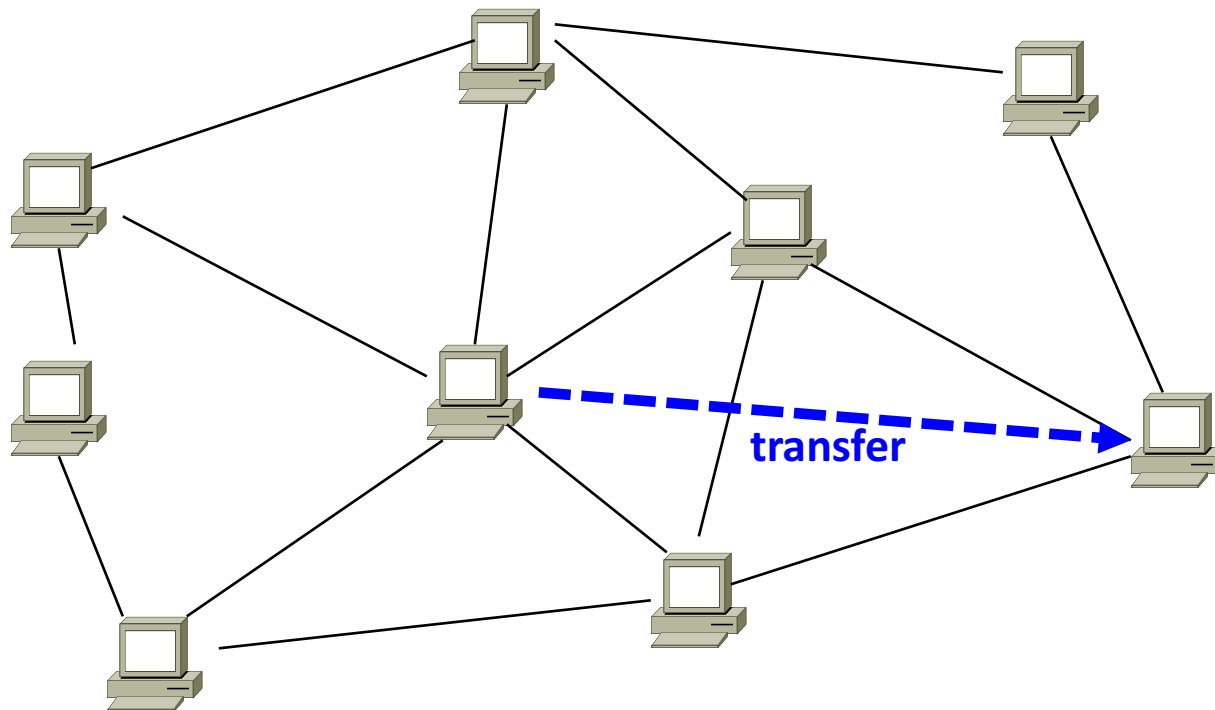
- ▶ Napster
 - ▶ Client-server search
 - ▶ “P2P” file xfer



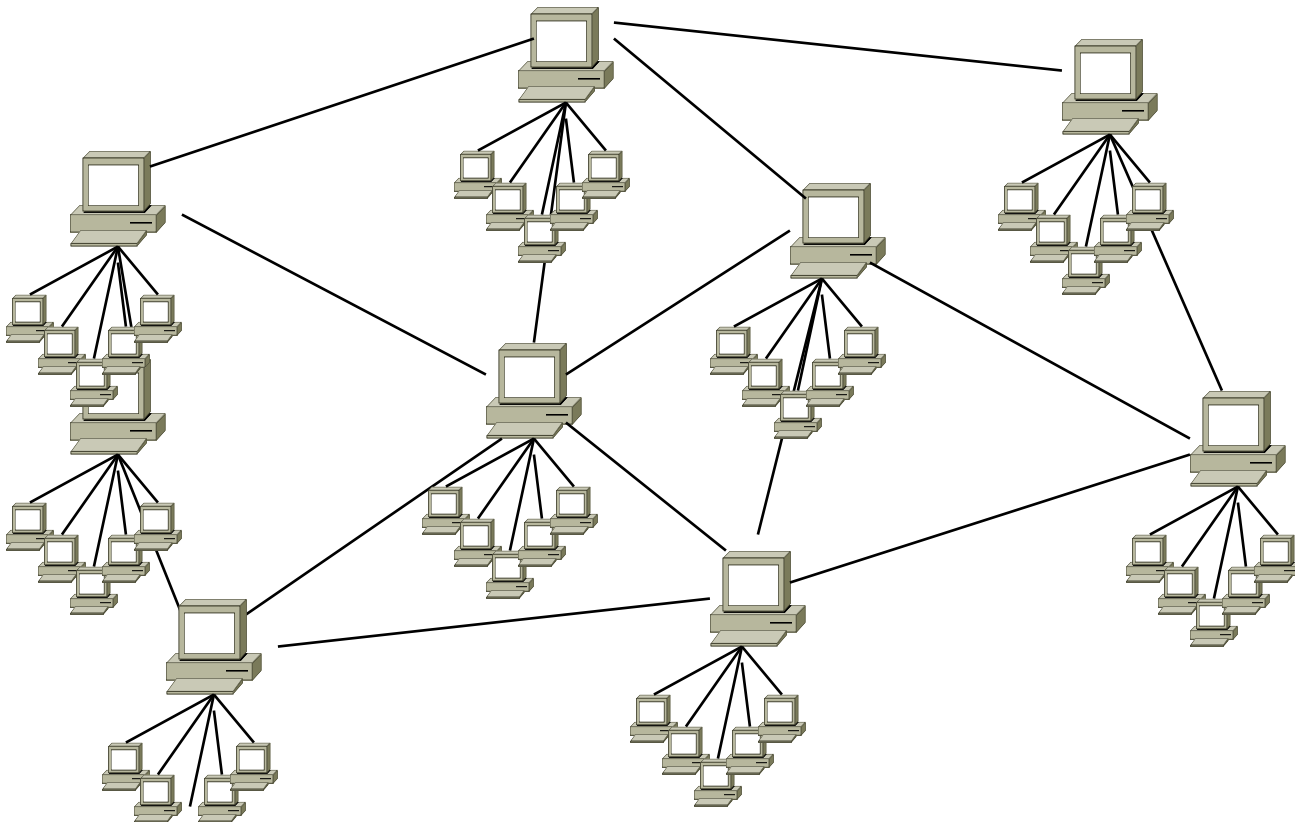




Direct transfer



- ▶ Kazaa/Gnutella
 - ▶ Gnutella: self-organizing





P2P: Overlay Networks

- ▶ P2P applications need to
 - ▶ Track identities & IP addresses of peers
 - ▶ May be many and may have significant churn
 - ▶ Route messages among peers
 - ▶ If you do not keep track of all peers, this is multi-hop
 - ▶ In sum: P2P systems need to maintain a network
 - ▶ To make operations sufficiently performant
- The OVERLAY network**
- ▶ P2P system creates this overlay differently, but
 - ▶ Peers doing both naming and routing
 - ▶ IP network becomes the low level transport



Some takeaways from experience

- ▶ Mediated approach performs well in many cases
 - ▶ Scale is not a problem
 - ▶ Reliability can be addressed with replication of some sort
 - ▶ Networking can be made resilient too.
- ▶ However...
 - ▶ Not always feasible: Performance not often key issue!
 - ▶ ... liability... tampering...



Some takeaways from experience

- ▶ Things that flood-based systems...
 - ▶ ...do well
 - ▶ Organic scaling
 - ▶ Decentralization of visibility and liability
 - ▶ Finding popular stuff
 - ▶ ...do poorly
 - ▶ Finding unpopular stuff
 - More flooding necessary
 - ▶ Vulnerabilities: data poisoning, tracking, etc.
 - ▶ Guarantees about anything (answer quality, privacy, etc.)



Structured Overlays
