Posrednici umreženih sustava

Prof. dr. sc Siniša Srbljić

Dr. sc. Ivan Benc

Fakultet elektrotehnike i računarstva,
Zavod za elektroniku, mikroelektroniku, računalne i inteligentne sustave

P2P - History

- Internet originally was a peer-to-peer system
  - Late 1960s, ARPANET
  - More open and free
  - Early killer apps: FTP and Telnet
    - client/server applications, but usage patterns were symmetric
    - This fundamental symmetry is what made Internet so radical
  - USENET, DNS

P2P History

- Switch to client/server
  - 1990s
  - Rise of the commercial Internet
  - NAT, Firewalls, Private IPs, Assymetric bandwith
  - Common usage patterns
    - Mostly downloading data
  - Exceptions
    - Email, News, Chat
    - But detailed manual and ISP instructions needed

P2P History

In the year **2000**, though, something has changed - or, perhaps, reverted. The network model that survived the enormous growth of the previous five years has been turned on its head. What was down has become up; what was passive is now active. Through the music-sharing application called **Napster**, and the larger movement dubbed "**peer-to-peer**," the **millions** of users connecting to the Internet have started using their ever more powerful home computers for more than just browsing the Web and trading email. Instead, machines in the home and on the desktop are connecting to each other **directly**, forming groups and **collaborating** to become user-created search engines, **virtual supercomputers**, and filesystems.

N.Minar and N.Hedlund: Peer to Peer - Harnessing the Power of Disruptive Technologies

P2P Definition #1

a **computer network** that relies primarily on the **computing power and bandwidth of the participants in the network** rather than concentrating it in a low number of servers

P2P Definition #2

collection of **heterogenous distributed** resources which are connected by the network

P2P Definition #3

opposite of client/server :-)

P2P Advantages

- Scalability
  - More users → more resources
- Survivability
  - There is no single point of failure
- Economics
  - Cheaper resources
- Bandwith
  - Bypass servers → direct connection

P2P Applications

- File Sharing
- Media Distribution
- Chat
- Data store
- Video Stream
- Telephony
- Forums
- Games

P2P Networks and Protocols

- BitTorrent
- Direct Connect
- Freenet
- Gnutella
- Napster
- JXTA
- eDonkey
- Kad network
- WPNP
- OpenNAP
- ...

P2P Middleware

- Why middleware?
  - Individual P2P networks
    - Lengthy and costly development
  - Rapid Application Development
  - Short time-to-market requirements
  - Example: project JXTA

JXTA

JXTA (Juxtapose)

- P2P platform and middleware
- Sun Microsystems 2001.
- Open source
- www.jxta.org

JXTA (Juxtapose)

- Set of open and generalized P2P protocols
- Enable any connected device on the network to communicate and collaborate as peers
- Independent of any programming language
- Multiple bindings exist for different environments

JXTA layers

- Three layers:
  - Core
    - Fundamental functionalities like transport and discovery
  - Service
    - Searching, indexing, file sharing, security ...
  - Applications

JXTA - Advertisements

- Fundamental concept in JXTA
- Advertisements represent resources
  - Peers, peer groups, services ...
- Distinguished by unique IDs
- Advertisement definitions are open XML documents

JXTA - Advertisements

- JXTA communication
  - Advertisement exchange in the network
- Advertisement caching for better performance
  - Optionally, not required by JXTA bindings
  - Cached advertisement have defined *lifetime*

JXTA – Peers and Peer groups

- Peer
  - Software running one or more JXTA protocols and
  - Data including name and unique *PeerID*
- Peer groups
  - All peers are members of the global *Net Peer Group*
  - May additionally join any number of groups
  - *Peer Groups* form subnets within the global JXTA network

JXTA – Services

- Peer services vs. Peer group services
  - Peer services
    - Single point of failure
  - Peer group services
    - Remain functional as long as peers stay present in the peer group
  - Core Services
    - Discovery, membership, access, pipe, resolver, monitoring

JXTA - Modules

- Modules implement services
  - Represent an abstraction of a piece of software or code
  - Not bound to any language
    - Allows individual services being implemented for multiple platforms
    - Can be Java classes, XML files, ...
  - All of JXTA built-in services are implemented using modules

JXTA - Protocols

- Protocols standardize the message-exchange by defining the message format and sequence
- Core protocols
  - Peer Resolver Protocol
  - Peer Discovery Protocol
  - Peer Information Protocol
  - Pipe Binding Protocol
  - Endpoint Routing Protocol
  - Rendezvous Protocol

JXTA - Messages

- JXTA's fundamental communication facilities are message-oriented and asynchronous
- Message types
  - Binary
  - XML encoded
- Applications usually use *pipes* to send and receive messages

JXTA - Pipes

- Pipes
  - virtual communication channels for message exchange
  - Hide network's complexity
  - Abstract from underlying protocols (TCP, HTTP ...)
  - Sending pipe endpoint = *output pipe*
  - Receiving pipe endpoint = *input pipe*

JXTA - Pipes

- Pipe types
  - Basic
    - simple point-to-point
    - Insecure and unreliable
  - Secure
    - virtual TLS (Transport Layer Security) connection for data encryption
  - Propagate pipe
    - multiple input and output endpoints

JXTA - Security

- JXTA does not guarantee that peers are the ones that they claim to be
  - Peers are not trustful
  - Trusted central authority lacks
- Encryption and signature
  - JXTA supports both secret key and public/private key encryptions

JXTA – Relay and Rendezvous peers

- Relay peers
  - Routing messages to peers / proxy
- Rendezvous peers
  - Propagate messages to all group members
  - Take care of discovery queries
  - Cache advertisements requested by connected peers

JXTA – Relay and Rendezvous peers

- Rendezvous peers
  - After peers join a group, they automatically seek rendezvous peer and
  - Establish permanent connection
  - If no rendezvous peers exist
    - Peers themselves become rendezvous peers
    - Serve others

JXTA – Review

- Shared environment
  - Dynamic peer groups creation = subenvironments
- Scalability
  - Large number of groups
  - Number of peers in a group is limited
- Dynamic network
  - Only rudimentarily considered

JXTA – Review

- Quality-of-Service
  - Not guaranteed
- Security
  - Trust not guaranteed
  - Encryption is supported

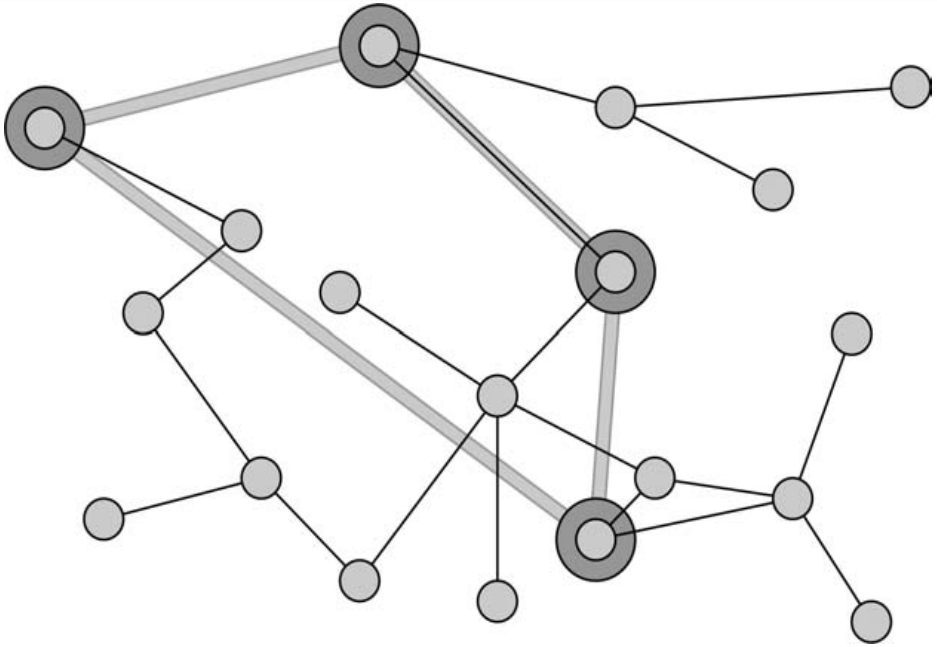
P2P Messaging System

P2P Messaging System

- Goal
  - Scalable, robust and fast middleware for P2P group communication
  - Fundamental concept: *multiring topology*
- Publish/Subscribe model
  - Topic-based groups
  - Publishers send messages
  - Subscribers receive messages

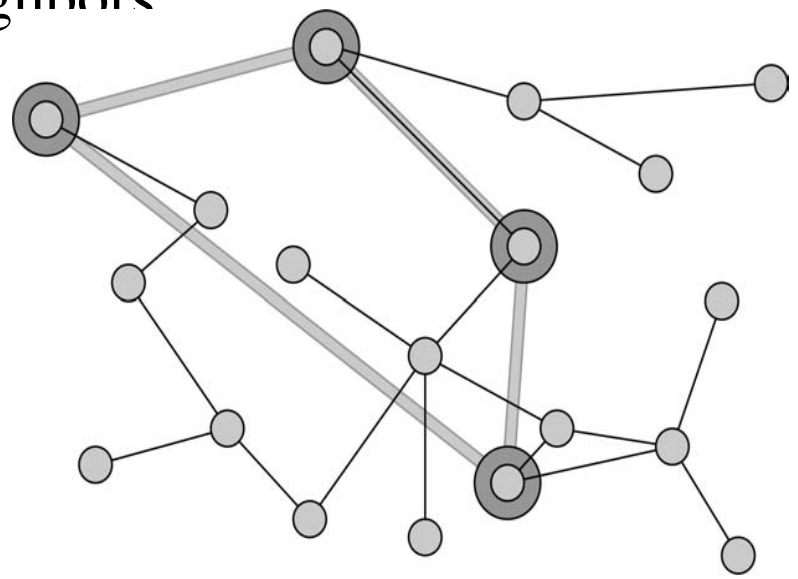
P2P Messaging System

- Virtual overlay networks
  - Dynamically created for each group
  - Groups cannot interfere with each other



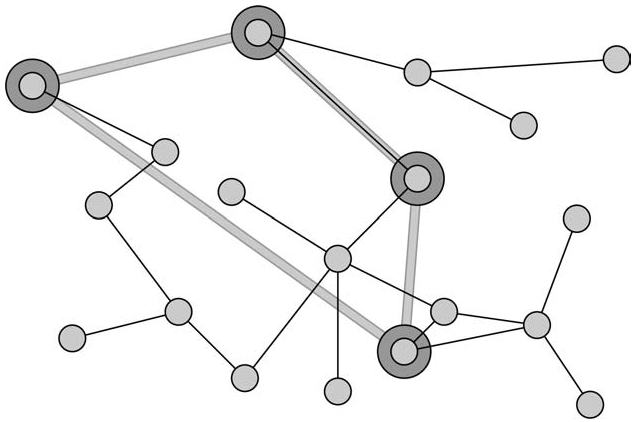
P2P Messaging System

- Multiring topology
  - Based on rings
  - Scalability
    - Workload of any node is independent of the total number of nodes
    - Each node depends only on its neighbors in both directions
    - Robust (manageability)



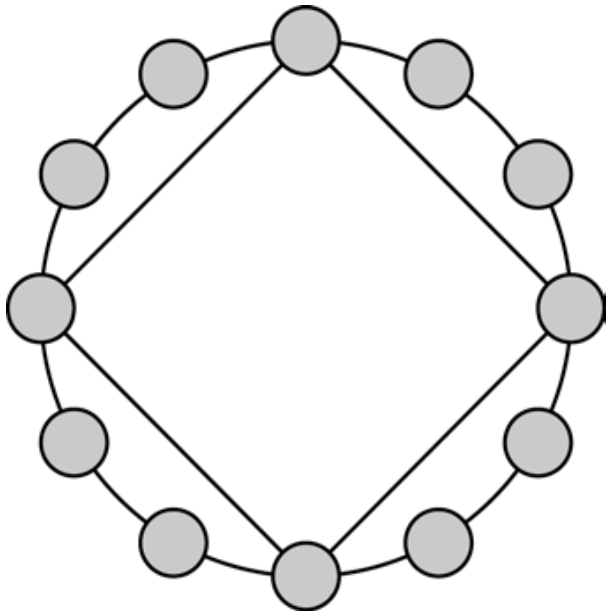
P2P Messaging System

- Multiring topology
  - Example: broadcast
    - Node sends message to its neighbors in both directions
    - Each node that receives message from one of its neighbors passes them on to its other neighbor
    - Arriving messages are stored in FIFO
    - Node ignores messages that arrived more than once



P2P Messaging System

- Multiring topology
  - Inner rings and and ratio *q*
    - Inner rings are shortcuts to distant ring sections and therefore decrease latency
    - Inner ring formation is recursive: inner rings can have further inner rings
    - Ratio *q*
      - Num of Inner ring nodes to
      - Num of outer ring nodes

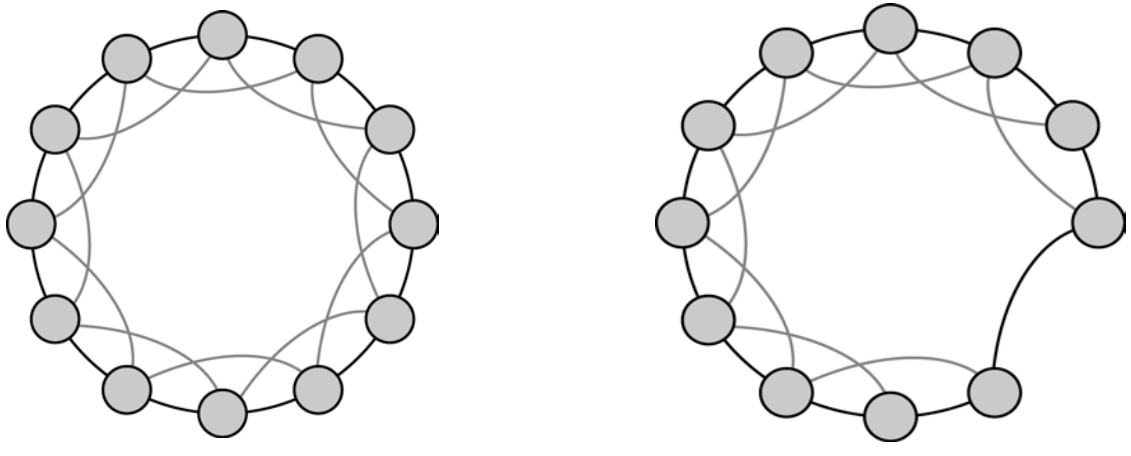


P2P Messaging System

- Multiring topology
  - Nodes periodically verify their bandwith
  - Compare the values with neighboring nodes
  - Node with superior bandwith is elected as an inner ring node
  - System dynamically balances inner ring nodes

P2P Messaging System

- Failure tolerance
  - Sending messages in two directions
  - Backup links
    - Each node has two backup links to the nodes next to its neighbors
    - When node fails, P2P MS removes broken links and activates backup links



P2P Messaging System

- Implicit Dynamic Routing
  - Problem: messages may arrive from several sources
  - Dual-link mode
    - Primary and Secondary links
    - Primary links allow immediate sending of messages
    - Secondary links must announce the availability of messages before sending
    - Messages are actually sent only if receiver needs them

P2P Messaging System

- Implicit dynamic routing
  - Dual-link mode
    - Initially all links are primary
    - Switch to secondary if a link delivers obsolete messages frequently
    - Receiver keeps track of the last y messages
    - Primary link: x of y new messages
    - Switching speed: x/y
    - Low values: adjust to changes quickly

P2P Messaging System

- Quality-of-Service (QoS)
  - Message priority
    - Key QoS parameter
    - High-priority messages are delivered before messages of lower priority
  - Preservation priority
    - Defines which messages will be preserved over others
    - Critical situations when message queues are full
      - System drops lower priority preservation messages first

P2P Messaging System

- Quality-of-Service (QoS)
  - Expiration time
    - Specifies when a message loses its relevance
      - System discards expired messages to reduce network traffic
  - Delivery mode
    - Indicates the main concern that delivery tries to satisfy
    - Modes: latency, throughput, redundancy optimization ...
    - System decides if a message is forwarded within inner rings depending on the delivery mode

P2P Messaging System

- Quality-of-Service (QoS)
  - Message filtering
    - Peers may have individual preferences
      - Sorting out irrelevant messages
    - Reduces network traffic
    - Backup and inner ring links are used to bypass peers
      - Implies that each peer knows about the message filters of its two neighbors
      - Worst case: none of the peers needs a message
        - This tehnique alone avoids half of the network traffic!
          - Message is delivered only to every second peer
        - Same concept with inner ring links compensates such scenarios

P2P Messaging System

- Review
  - Shared environment
    - Dynamic overlay networks creation, subenvironments
  - Scalability
    - Scalable multiring topology
    - Dynamically adjust to changes
  - Dynamic network
    - Reduced dependancy on single nodes
    - New routes evolve dynamically
    - Backup links compensate node failures and congestions

P2P Messaging System

- Review
  - Dynamic node characteristics
    - Nodes relocate within the topology according to their characteristics
  - Network heterogeneity
    - Positions within the topology are assigned according to individual peer capabilities
  - Quality-of-Service
    - Not guaranteed
    - Delivery process can be influenced by QoS parameters

P2P Messaging System

- Review
  - Security
    - Decentralized networks lack trust
    - Security is deferred to centralized service providers
    - Encryption depends on the underlying network module

Hybrid Middleware

Hybrid Middleware

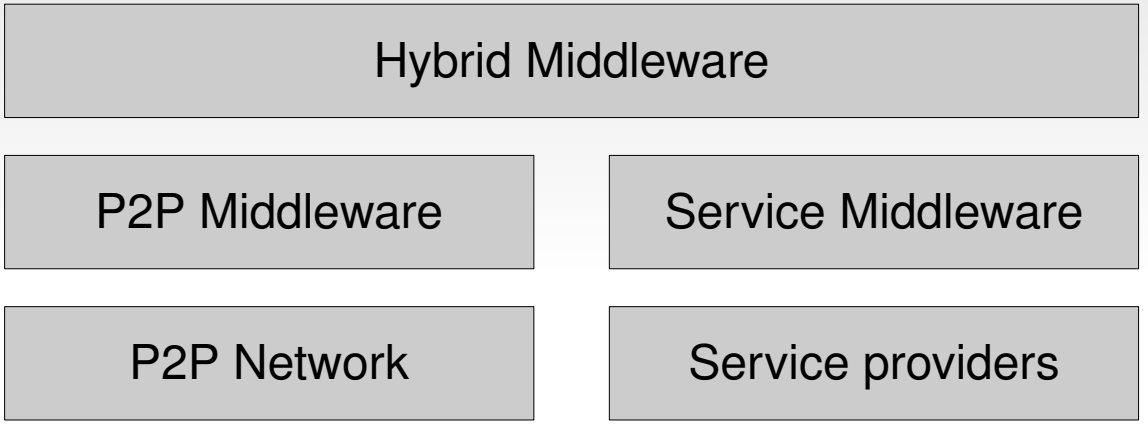
- P2P
  - High scalability
  - High resource availability
  - High fault tolerance
  - Self-organizing architecture
  - Low infrastructure costs
  - No global storage
  - No control
  - Not trusted
  - No legacy system integration
- Centralized network
  - Limited scalability
  - Limited resource availability
  - Limited fault tolerance
  - Need setup and administration
  - High infrastructure costs
  - Global storage
  - Control
  - Trusted
  - Legacy/Enterprise system integration

Hybrid Middleware

- Combines advantages of both P2P and centralized networks
- Uses different distributed computing concepts and topologies
  - Individual task requirements
  - Different from Napster-like P2P middlewares
    - Napster has fixed assignment of tasks

Hybrid Middleware

- Abstract model



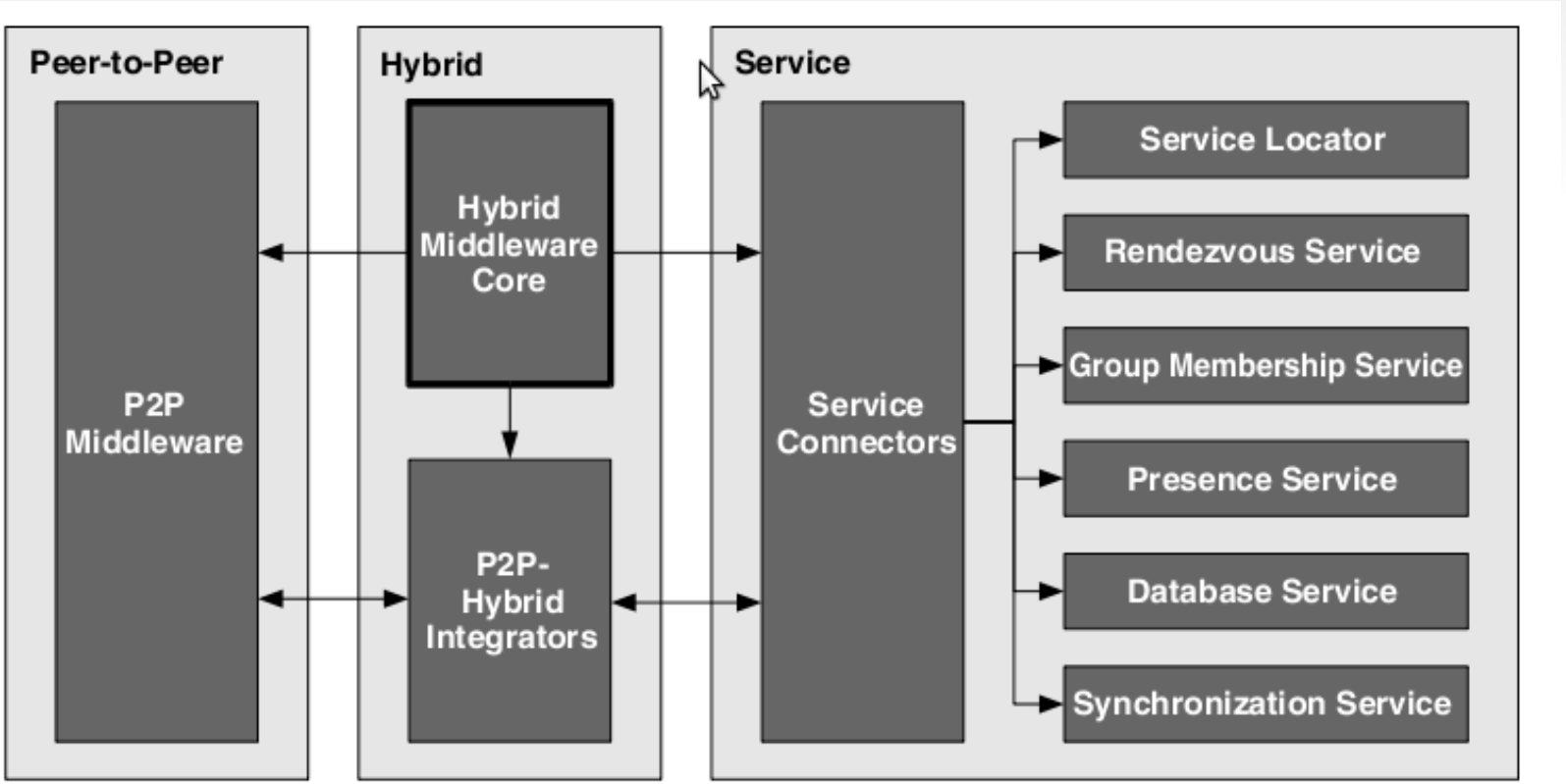
- P2P Network and Service Providers
  - Network layer
- P2P and Service Middleware
  - Existing middleware for accesing network

Hybrid Middleware

- Service Providers
  - Centralized service
    - Server, cluster ...
  - Higher availability
  - Easier to locate
  - Dedicated server role
  - Limited number of nodes

Hybrid Middleware

- Conceptual Model and Services



Hybrid Middleware

- Conceptual Model and Services
  - All services are accessed by Service Connectors
  - If the functionality does not involve P2P
    - Hybrid systems directly access Services
  - P2P-Hybrid Integrators
    - Provide functionality that depends on both P2P Middleware and Service Providers
  - Hybrid system may also delegate functionality to the P2P middleware directly
    - eg. when no depenedencies to Services exist