# Peer-to-Peer Applications

(Version February 27, 2023)

Xiao Mingzhong

CIST, Beijing Normal University

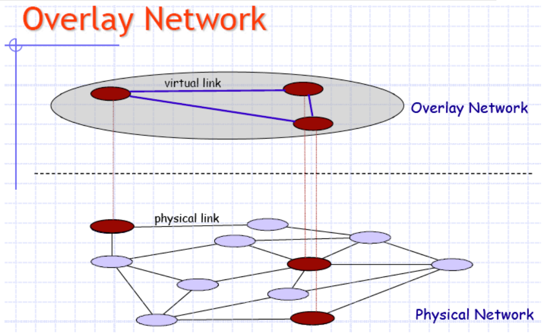
## Outline

- □ C/S vs P2P
- □ Applications
  - Napster
  - Gnutella
  - BT
- □ DHT technique
  - Chord
- □ P2P Issues

## The alternative: P2P Model

- □ Quickly grown in popularity
  - Hundreds of file sharing applications
  - About 31% of all Internet traffic is due to BitTorrent
  - Audio/Video transfer now dominates traffic on the Internet
- □ P2P "takes advantage of resources at the edges of the network" (Clay Shirky, O'Reilly)
  - End-host resources have increased dramatically
  - Broadband connectivity now common
  - Resources can be: processing cycles, storage space,bandwidth,data, Wifi密码(wifi万能钥匙) …

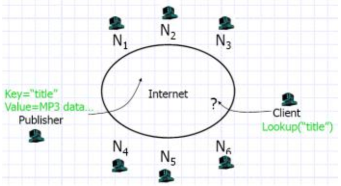
## P2P App forms a Overlay network



结构化和非结构化网络指的是网络拓扑是否是有组织地: 命名和寻址各为何?

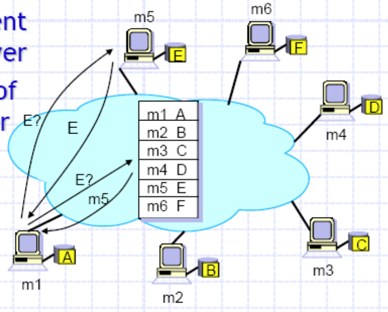
## What we will concentrate on

- □ Retrieving resources is a fundamental issue in P2P systems due to their inherent geographical distribution
  - The problem is to direct queries towards nodes that can answer them in the most efficient way
  - e.g., file sharing app.



## Napster

- ✖ Join: on startup, client contacts central server
- ✖ Publish: reports list of files to central server
- ✖ Search: query the server => return someone that stores the requested file
- ✖ Fetch: get the file directly from peer



## About Napster…

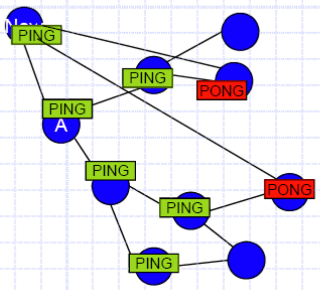
- □ Many researchers argued that it is not a pure P2P system
  - It is a P2P system since allows small computers on edges to contribute
  - All peers are active participants as service provider not only as consumer
- □ PROs: simple and search scope is O(1)
- □ CONs: server maintains O(N) state and does all processing; sigle point of failure;

## Gnutella

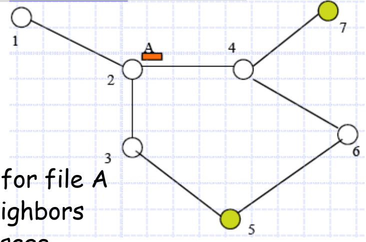
- ✖ Query Flooding:
  - Join: on startup, client contacts a few other nodes; these become its “neighbors”
  - Publish: no need
  - Search: ask neighbors, who as their neighbors, and so on... when/if found, reply to sender.
  - Fetch: get the file directly from peer

## Gnutella: joining the network

- The new node connects to a well known 'Anchor' node.
- Then sends a PING message to discover other nodes.
- PONG messages are sent in reply from hosts offering new connections with the new node.
- Direct connections are then made to the newly discovered nodes.



## Gnutella search mechanism



1) Node 2 initiates search for file A
2) Sends message to all neighbors
3) Neighbors forward message
4) Nodes that have file A initiate a reply message
5) Query reply message is back-propagated
6) File download

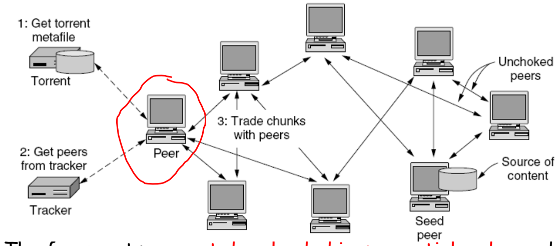
## Gnutella: Pro and Cons

- □ PROs:
  - Fully de-centralized
  - Search cost distributed
  - "search for S" can be done in many ways, e.g., structured database search, simple text matching, "fuzzy" text matching, etc.
- □ CONs:
  - Flood of requests.
  - Search scope is O(N);
  - Nodes leave often, network unstable

## Bittorent(以文件为管理对象,块为传输对象)

- □ Focused on efficient fetching, not searching:
  - Distribute the same file to all peers
  - Single publisher, multiple downloaders
- □ BT terminology
  - Torrent: a meta-data file describing the file(s) to be shared. A .torrent file holds:Names of the files,Sizes,Checksum of all blocks,Address of the tracker,Address of peers
  - Seed: a peer that has the complete file and still offers it for upload
  - Leech: a peer that has incomplete download
  - Swarm: all seeders/leeches together make a swarm
  - Tracker: a server that keeps track of seeds and peers in the swarm and gathers statistics. When a new peer enters the network, it queries the tracker to provide a list of peers.

## BT: content distribution



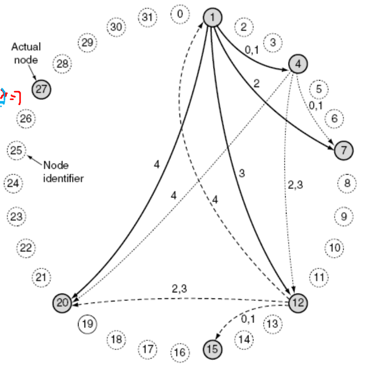
- □ The fragments are not downloaded in sequential order and need to be assembled by the receiving machine
- □ Clients start uploading what they already have before the whole download is finished
- □ Tit-for-tat: gives peers incentive to share resources

## DHT:集中式索引P2P化

- □ Tracker server 是一个集中式的索引点！ 一个集中式的索引信息是否能像DNS信息一样分布存储和检索呢？
  - ■ 索引P2P化的想法---有组织地
    - 每个Peer负责存储管理部分信息→怎样分工？
    - 每个Peer都能快速找到需要的索引→P2P网络中请求如何快速到达责任节点？路由问题。
    - 节点来去自由的网络中，如何使索引查询服务快速可靠？→移交问题。
  - ■ DHT： Chord,CAN,Pastry,Tapestry,Kademlia
    - 结构化和非结构化P2P网络：后者的网络邻居可以是任何可能的Peers,而前者需遵循一定的策略.

## Chord环

- □ Node ID: SHA1
- □ Actual node
- □ Successor(6)=7 S(7)=7
- □ Successor(SHA1 (content)) saves the content INDEXs
- □ How to routing?
  - ■ 沿环找继任
  - ■ e.g., at node 1, find key=16.
  - ■ 线性搜索
- □ Finger table加速
  - ■ Start=k+2^I
  - ■ 找最接近的前任的继任 (16:12,15,20)
  - ■ Log（N）搜索



| Start | IP addr of successor |
|---|---|
| 2 | 4 |
| 3 | 4 |
| 5 | 7 |
| 9 | 12 |
| 17 | 20 |

Node 1's finger table

| Start | IP addr of successor |
|---|---|
| 5 | 7 |
| 6 | 7 |
| 8 | 12 |
| 12 | 12 |
| 20 | 20 |

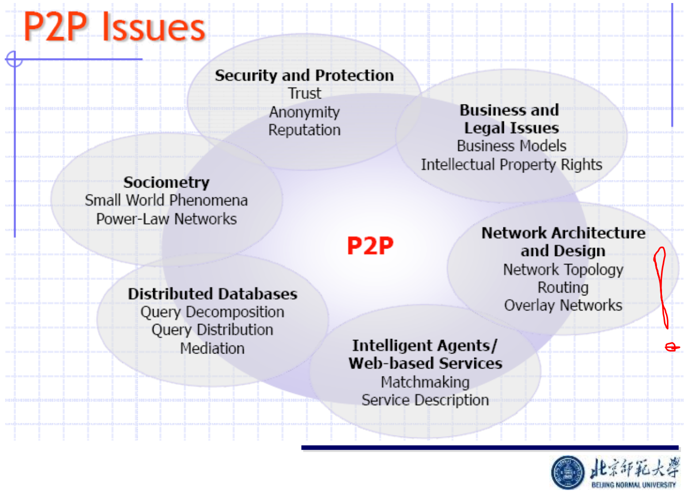
Node 4's finger table

| Start | IP addr of successor |
|---|---|
| 13 | 15 |
| 14 | 15 |
| 16 | 20 |
| 20 | 20 |
| 28 | 1 |

Node 12's finger table

## Chord ring: join and leave

- □ Join
  - ■ 找自己的后继的IP地址（因为“是代表”），并获悉前任节点；
  - ■ 通知双方自己要加入环，各自修改“前后任”信息；
  - ■ 后继移交本该加入节点负责管理的索引信息
  - ■ 后台定期调用successor更新指取表
- □ Leave
  - ■ 温和离开：权力移交给继任；通知前任离开→前任更新后继；
  - ■ 突然离开：索引多份存储解决索引丢失；第1，2，3，...后继方案应对；
- □ 为何实用的是Kademlia？



# P2P Issues

P2P

**Security and Protection**
Trust
Anonymity
Reputation

**Business and Legal Issues**
Business Models
Intellectual Property Rights

**Sociometry**
Small World Phenomena
Power-Law Networks

**Network Architecture and Design**
Network Topology
Routing
Overlay Networks

**Distributed Databases**
Query Decomposition
Query Distribution
Mediation

**Intelligent Agents/ Web-based Services**
Matchmaking
Service Description

北京師範大學
BEIJING NORMAL UNIVERSITY