Introduction to Peer-to-Peer Networks

- The Story of Peer-to-Peer
- The Nature of Peer-to-Peer: Generals & Paradigms
- Unstructured Peer-to-Peer Systems
- Sample Applications
A Peer-to-Peer system is a self-organizing system of equal, autonomous entities (peers) which aims for the shared usage of distributed resources in a networked environment avoiding central services.

Andy Oram
The Old Days

- **NetNews (nntp)**
  - Usenet since 1979, initially based on UUCP
  - Exchange (replication) of news articles by subscription
  - Group creation/deletion decentralised

- **DNS**
  - Distributed delegation of name authorities:
    - file sharing of host tables
  - Name “Servers” act as peers
  - Hierarchical information space permits exponential growth

- Systems are manually configured distributed peers
SETI@home: Distributed Computing

- Search for Extraterrestrial Intelligence (SETI)
- Analyse radio signals from space
- Globally shared computing res.
- Idea 1995
- First version 1998
- 2002 ≈ 4 Mio clnt
- E.g. Screensaver
- http://setiathome.berkeley.edu/ - ongoing

From Anderson et. al.: SETI@home, Comm. ACM, 45 (11), Nov. 2002
SETI@home (2)

- http-based client-server model
- No client-client communication
- Data chunks: load & return
- N-redundancy for fault detection
- Attacks: bogus code theft of email addresses

From Anderson et. al.: SETI@home, ibidem, Nov. 2002
Napster

- **MAY 1999: Disruption of the Internet community**
- First Generation of File sharing: Introduction of Napster
  - Users not only consume and download, but also offer content
  - Users establish a virtual network, entirely independent from physical network and administrative authorities or restrictions
  - Basis: UDP and TCP connections between the peers
- **Napster provides centralised indexing**
  - Clients upload their file list to Napster Server
  - Clients query Index Server and receive full provider list
- **Data exchange directly between peers**
Napster

- December 1999: RIAA files a lawsuit against Napster Inc.
- March 2000: University of Wisconsin reports 25% of its IP traffic is Napster traffic
- February 2001: 2.79 billion files per month exchanged via Napster
- July 2001: Napster Inc. is convicted
  - Target of the RIAA: the central lookup server of Napster
  - Napster has to stop the operation of the Napster server
  - Napster network breaks down
- Napster failed (technically & legally) at its single server point.
Gnutella

- File sharing fully decentralised
- Open source software
- March 2000: Release 0.4 – with network flooding
- Spring 2001: Release 0.6 – improved scalability
Gnutella 0.4

- Pure P2P system – no central indexing server
- Operations:
  1. Connect to at least one active peer (address received from bootstrap)
  2. Explore your neighborhood (PING/PONG)
  3. Submit Query with a list of keywords to your neighbors (they forward it)
  4. Select “best” of correct answers (which we receive after a while)
  5. Connect to providing host/peer
- Scaling Problems due to network flooding
Gnutella 0.4: How Does It Work
Basic Routing Behavior

- **Request messages:**
  - Include a hop-counter, a GUID and a TTL (Time-to-Live) in the header
  - TTL determines along how many hops a message may be forwarded
  - Are flooded in the overlay network
    - Every node forwards every incoming message to all neighbors except the neighbor, it received the message from
  - Request messages terminate, if
    - Same message-type with same GUID is received more than once (loop!!)
    - Hop-counter=TTL

- **Response messages:**
  - Include a hop-counter, a GUID and a TTL (Time-to-Live) in the header
  - GUID is the same as of the initializing request message
  - Are routed back on the same way to the requestor, the request message had been received
    - every peer has to store the GUID of each request for a certain amount of time
    - No flooding to save resources
Gnutella 0.6

- Hybrid P2P System – Introduction of Superpeers
- Improved scalability: signalling reduced to Superpeers
- Election mechanism decides which node becomes a Superpeer or a Leafnode (depending on capabilities (storage, processing power) network connection, the uptime of a node,...)
- Leafnodes announce their shared content to the Superpeer they are connected to
- Superpeers carry local routing tables
Gnutella 0.6: How Does It Work

From:
J. Eberspächer, R. Schollmeier: First and Second Generation Peer-to-Peer Systems, in LNCS 3485
The Gnutella Network

Measurements from May 2002

From:
J. Eberspächer, R. Schollmeier: *First and Second Generation Peer-to-Peer Systems*, in LNCS 3485
Impacts of P2P at the Abilene Backbone

- Unidentified + data_transfers + file_sharing causes 90% of the traffic
- Unidentified traffic and data_transfers increased significantly
  - Parts of P2P is hidden (port hopping,...)
  - Some P2P applications use port 80 → data_transfers

Data source: http://netflow.internet2.edu/weekly/
The Nature of P2P

- P2P Networks overlay network infrastructure
- Implemented on application layer
- Overlay Topology forms a virtual signaling network established via TCP connects
- Peers are content provider + content requestor + router in the overlay network
- Address: General Unique ID
P2P & Distributed Systems Paradigm

- Coordination among equal components
- Decentralised & self organising
- Independence of individual peers
- Scalability over tremendous ranges
- High dynamic from volatile members
- Fault resilience against infrastructure & nodes
- Incentives instead of control
P2P & Internetworking Paradigm

- Loose, stateless coupling among peers
- Serverless & without infrastructural entities
- Dynamic adaptation to network infrastructure
- Overcome of NATs or port barriers
- Client-Server principle reduced to communication programming, not an application paradigm anymore
- Somewhat “Back to the Internet roots”:
  - Freedom of information
  - Freedom of scale
  - But: Freedom of Internet infrastructure & regulation

But: Freedom of Internet infrastructure & regulation

http://www.informatik.haw-hamburg.de/~schmidt
### Client-Server vs. Peer-to-Peer

<table>
<thead>
<tr>
<th><strong>Client-Server</strong></th>
<th><strong>Peer-to-Peer</strong></th>
</tr>
</thead>
</table>
| 1. Resources are shared between the peers  
2. Resources can be accessed directly from other peers  
3. Peer is provider and requestor (Servent concept) | |

### Unstructured P2P

<table>
<thead>
<tr>
<th><strong>1st Generation</strong></th>
<th><strong>2nd Generation</strong></th>
</tr>
</thead>
</table>
| 1. Server is the central entity and only provider of service and content.  
    $\rightarrow$ Network managed by the Server  
2. Server as the higher performance system  
3. Clients as the lower performance system | |

#### Example: WWW

**Centralized P2P**
- 1. All features of Peer-to-Peer included
- 2. Central entity is necessary to provide the service
- 3. Central entity is some kind of index/group database

**Pure P2P**
- 1. All features of Peer-to-Peer included
- 2. Any terminal entity can be removed without loss of functionality
- 3. $\rightarrow$ No central entities

**Hybrid P2P**
- 1. All features of Peer-to-Peer included
- 2. Any terminal entity can be removed without loss of functionality
- 3. $\rightarrow$ dynamic central entities

**DHT-Based**
- 1. All features of Peer-to-Peer included
- 2. Any terminal entity can be removed without loss of functionality
- 3. $\rightarrow$ No central entities
- 4. Connections in the overlay are “fixed”

**Examples:** Gnutella 0.4, Freenet, JXTA, Chord, CAN

---

From: J. Eberspächer, R. Schollmeier: *First and Second Generation Peer-to-Peer Systems*, in LNCS 3485

Hamburg University of Applied Sciences
Unstructured Peer-to-Peer Systems

- Decentralized and self organizing (with possible centralized elements)
- Content:
  - Distributed “randomly” on the network, with several replicas
  - Content and its descriptions are not structured (stays at the nodes which bring it into the network)
- Content transfer:
  - Out of band, i.e. on separate connections and not via signaling connections
  - Mostly via HTTP
- Generally two kinds of requests:
  - Content requests: to find content in the overlay
  - Keep-alive requests: stay connected in the overlay
- Initially developed for file-sharing
- Various realizations exist
Basic Characteristics of Centralized P2P

- **Bootstrapping**: Bootstrap-server = central server
- Central entity can be established as a server farm, but one single entry point = single point of failure (SPOF)
- All signaling connections are directed to central entity
- Peer ↔ central entity: P2P protocol, e.g. Napster protocol
  - To find content
  - To log on to the overlay
  - To register
  - To update the routing tables
  - To update shared content information
- Peer ↔ Peer: HTTP
  - To exchange content/data:
Centralized P2P Routing
Basic Characteristics of Pure P2P

- **Bootstrapping:**
  - Via bootstrap-server (host list from a web server)
  - Via peer-cache (from previous sessions)
  - Via well-known host
  - No registration

- **Routing:**
  - Completely decentralized
  - Reactive protocol: routes to content providers are only established on demand, no content announcements
  - Requests: flooding (limited by TTL and GUID)
  - Responses: routed (Backward routing with help of GUID)
Pure P2P Routing
Basic Characteristics of Pure P2P (2)

- Signaling connections (stable, as long as neighbors do not change):
  - Based on TCP
  - Keep-alive
  - Content search

- Content transfer connections (temporary):
  - Based on HTTP
  - Out of band transmission
Model of Pure P2P Networks

Degree distribution:

\[ p(d) = \begin{cases} 
  cd^{-1.4}, & 0 < d \leq 7 \\
  0, & \text{in any other case} 
\end{cases}, \quad \text{with } c = \left( \sum d p(d) \right)^{-1} \]

average: \( \bar{d} = 2.2 \)

\( \text{var}(d) = 1.63 \)

According Sample Graph:

Separated subnetworks

Major component
Basic Characteristics of Hybrid P2P

- **Bootstrapping:**
  - Via bootstrap-server (host list from a web server)
  - Via peer-cache (from previous sessions) or well-known host
  - Registration of each Leafnode at the Superpeer it connects to, i.e. it announces its shared files to the Superpeer

- **Routing:** Partly decentralized
  - Leafnodes send request to a Superpeer
  - Superpeer distributes this request in the Superpeer layer
  - If a Superpeer has information about a matching file shared by one of its leafnodes, it sends this information back to the requesting leafnode
  - Hybrid protocol (reactive and proactive): routes to content providers are only established on demand; content announcements from leafnodes to their Superpeers
  - Requests: flooding (limited by TTL and GUID) in the Superpeer layer
  - Responses: routed (Backward routing with help of GUID)
Hybrid P2P Routing
Model of Hybrid P2P Networks

Degree distribution:

\[
p(d) = \begin{cases} 
    c d^{-1.4}, & 1 < d \leq 7 \\
    c 1^{-1.4} - 0.05, & d = 1 \\
    c 0.05, & d = 20 \\
    0, & \text{in any other case}
\end{cases}, \quad \text{with } c = \left( \sum_d p(d) \right)^{-1}
\]

average: \( \bar{d} = 2.8 \)

\[ \text{var}(d) = 3.55 \]

According Sample Graph:

- **Major component**
- **Separate sub networks**
- **Hub connections (2nd hierarchy)**
- **Superpeer**
- **Leafnode**
Abstract network structure of a part of the Gnutella network (222 nodes) Geographical view given by Figure on the right, measured on 01.08.2002

Geographical view of a part of the Gnutella network (222 nodes); The numbers depict the node numbers from the abstract view (Figure on the left, measured on 01.08.2002)

- Virtual network not matched to physical network. See path from node 118 to node 18.
- Superpeer (hub) structure clearly visible in abstract view
P2P Application Areas

- File sharing
- Media Conferencing
- Overlay Multicast: IPTV ...
- Resource Sharing: Grids
- Collaborative Communities
- Content based networking: e.g. Semantic Nets
- Mobile Adhoc Networks: e.g. Vehicular Communication
- De-personalization tools: e.g. Tor
- Inspiration for a next generation Internet
- ...
File Sharing: BitTorrent

**Peer discovery:** tracks peers in torrent

**torrent discovery:** search for torrents; provides .torrent file

**torrent:** group of peers exchanging chunks of a file

trading chunks
BitTorrent „Eco“-System

Simple Interface:
- Publishing – .torrent metainfo file + Tracker
  - Tracker provides download peers
  - Trackerless clients use distributed indexing
- Downloading – use BitTorrent via a Web browser
  - Uploading is started automatically

File exchange incentive:
- Tit-for-tat trade – balance upload and download connection-wise
Skype

- VoIP conferencing system
- Released 2003
- Central login server
- Hybrid P2P system otherwise
- Main focuses:
  - Detect users
  - Traverse NAT & Firewalls (STUN)
- Elects Superpeers according to network connectivity
- Uses Superpeers as relays
IPTV: The Video Tsunami

Video Road Hogs Stir Fear of Internet Traffic Jam

By STEVE LOHR
Published: March 13, 2008

Caution: Heavy Internet traffic ahead. Delays possible.

For months there has been a rising chorus of alarm about the surging growth in the amount of data flying across the Internet. The threat, according to some industry groups,
Resume

- P2P technologies offer an innovative overlay infrastructure for decentralized and distributed systems
- Due to the distributed nature, the signaling load is very high.
- Signaling load may be decreased by further structures
- Advantages:
  - Simple basic principle
  - Enhanced reliability
  - Redundancy (high replication rate)
  - Unsusceptible against Denial of Service attacks (DOS)
  - No single point of failure
- Problem: Increasing struggle with ISPs
References

