

# Peer-to-Peer Computing

## P2P

Per Brand

DSL-lab

SICS

# Rise and Fall of P2P in Media?

‘Peer-to-peer is the next great thing for the Internet’

Stanford Law Net Guru - Lawrence Lessig, 2000

‘Peer-to-peer computing is leading us into the third age of the Internet’

Bob Knighton - Intel Corp. Fall 2000

‘Is P2P plunging off the deep end?’

Wall Street Journal April 2, 2001

‘Does Peer-to-Peer Suck?’

Jon Katz - Slashdot - on reviewing P2P book. April 4, 2001

# Overview of talk

- What is peer-to-peer computing
- What makes it distinctive
- What are the potentials
- Examples of existing applications
- Possible applications
- What are the technical challenges
- Research in the area at SICS
- Speculations
- Summary

# Not covered in the talk

- If/how to make money from P2P- depends on
  - what becomes popular
  - what people are willing to pay for some service
  - what comes for free
- Don't ask a researcher
  - consider e-mail - more money has probably been made selling e-mail address lists than providing the service itself

# What is peer-to-peer computing

Webster definition of peer

‘one that is of equal standing with another’

Computing between equals

# What makes it distinctive

- Class of applications that take advantage of resources - storage, cpu, content, human presence - available at the edges of the network.
- Edge of the network
  - Users and their PCs and devices
  - Often
    - without permanent IP address
    - turned off

# The Contrast: Server-Centric

- The client/server paradigm
- The client is basically a glorified I/O device
- Computation, control, and information is kept at the server
  - Simpler to build centralized versus decentralized systems
  - Good, keep it simple if it is enough

# Litmus test

- From Clay Shirky
  - PCs are the ‘dark matter of the Internet’
- Application is P2P if
  - allows for variable connectivity and temporary network addresses
  - gives the nodes at the edges of the network significant autonomy

# Another characterization

- Peers in relation to each other act as
  - Clients (consumer of service) AND
  - Servers (producer of service) AND
  - Routers (in a virtual network) AND
  - Caches

# What makes (made) it so hyped

- Industry looking for something positive after .com death
- Has large, rather apparent, social consequences - makes for great interest
- Some very interesting applications became widely known and used
  - Napster, Gnutella, FreeNet

# What makes it important

- Technology dimension
  - Client/server technology has limitations
  - Peer-to-peer technology has different limitations
    - With both more can be achieved
- Social dimension
  - The Internet has already changed society
  - We can expect further changes

# Is P2P new?

- P2P is not new
  - Earlier (basic) Internet services more peer-to-peer than later ones
    - Early services:
      - e-mail, usenet, ftp, dns (hybrid)
    - Later services:
      - ISP, dynamic IP, firewalls, NAT
- Work on P2P technology
  - has been ongoing the whole time
  - by other names (until recently)

# Potential of P2P

- Better resource utilization
- Scalability (sometimes)
- Fault-tolerance (sometimes)
  - these 3 aspects are technical, and might never be seen by the end-user
- Improved automation in certain services
- Privacy
  - these 2 aspects will be very visible to the end-user

# Better resource utilization

- Estimate of Net-connected PCs
  - 10 billion megahertz of CPU
  - 10000 Terabytes of storage
  - How much is unused?
- Resources are available at the edge of the Internet
  - Not so much in the middle (servers, clusters)
  - Today often wasted
- P2P utilization ideas
  - Share resources among friends
  - Give resources to worthy causes
  - Sell/buy resources

# Scalability

- Technique for greater scalability on application level is
  - more cpu-power and more disk and split up the work
  - but communication overhead can be prohibitive
  - example, extending a cluster farm with more machines with high-speed local network
    - expensive for many types of applications that don't need this
  - servicing requests for replicated data is fairly easy to distribute
    - make use of free resources

# Fault-tolerance

- The key to fault-tolerance is redundancy
- In many cases needs tightly synchronized redundancy
  - fast, reliable, secure connections necessary
  - P2P does not help (rather the opposite)
- If loosely synchronized redundancy meets the bill then P2P helps.
  - Example: providing seldom updated information.
  - Copies available in many peers.
- Communication path redundancy in virtual network

# Improved Automation

- Peers expose what they do normally to other peers (that they trust)
  - basis for adding services without any extra work by people
  - example: music directory made available and database enlarged by downloading (Napster)
  - providing community service automatically from ordinary usage (selfishness)
    - **Tragedy of the Commons vs.**
      - **Example: e-mail spam**
    - **Cornucopia of the Commons**
      - **Example: Napster**

# Privacy

- Open sharing between peers
  - But not exposing to businesses/government etc.
    - what I browse, what I write, what I download, etc.
- Anonymous sharing
  - Not exposing to anyone the identity
    - of the author
    - the reader (download)
- Avoiding
  - Censorship (Free Haven)
  - Laws (Napster, Gnutella, FreeNet)
- P2P = Power 2 People

# Example 1: Napster

- Hybrid between peer-to-peer and server-centric
  - Napster centralized database holds the information where music is found
  - Music stored at peers (users)
  - Many copies of the same piece
- Motivation for architecture
  - Privacy - avoiding copyright laws, whoops!
  - Resource utilization
    - database at Napster smaller as songs stored elsewhere
    - probably not that important
  - Automation
    - participation automatic - once you download you're in the database
    - automatic participation in redundancy (fault-tolerance)

# Example 2: SETI

- Letting use share their resources in the search for extraterrestrial intelligence
- Motivation for architecture
  - Resource utilization
    - using those unused CPU resources at the edge of the network

# Example 3: Gnutella

- Sharing information among peers
- Pure P2P, completely decentralized
  - Napster without Napster
- Virtual network
  - dynamic, open
  - ad-hoc backbone (adapts to bandwidth differences)
  - for search and discovery
  - broadcast queries
- More anonymous than Napster but not downloads
  - Wall of Shame (downloads of entries with suspect keys - e.g. childporn)
  - Countermeasures

# Examples of Secrecy

- Sharing information in peer groups under various secrecy requirements
- Red Rover - keeping secret that you have a secret
  - targeted against authoritarian regimes
- Free Net
  - anonymous publication
  - virtual network + dropping non-used information
- Free Haven
  - anonymous persistent publication, resists tampering, censorship
  - mechanisms for accountability

# Key techniques - 1

- Virtual network
  - Path redundancy
  - Dynamic
    - users connect/disconnect
    - coping with firewalls (e.g. HTTP-tunneling)
    - coping with dynamic IP addresses
- Encryption
  - Public/private key
- Minimizing Information
  - Any one potential malicious peer has limited information
  - Example: protecting the path - each node has only access to the next node in the chain (rest encrypted)

# Key techniques - 2

- Obfuscation
  - padding
  - changing timings
  - dummy messages
- Reputation
- Frequent revision
- Micropayments
  - Nonfungible - non-redeemable
    - make user compute moderately hard computation problem before gaining access to resource
    - consider the peer-to-peer application: e-mail
  - Fungible - redeemable
    - electronic cash

# Examples of Attacks

- Physical - i.e. destroy a peer
  - replication
- Legal - close peer by legal action
  - replication, plausible deniability, make use of different legal systems in different countries
- Denial of Service
  - client puzzles
- Flooding
  - reputation system
- Infiltration
  - see to that it that the infiltrator has done more good in building a good reputation than he can do bad when he has it.

# Technical Challenges

- There are difficulties
- It's no accident that most services developed in last 5 years are client/server - it's easier
- Server technology behind the scenes
  - For scalability, fault-tolerance need clusters
  - Distributed computing research community
  - Decentralized in the meaning of distributed
  - Centralized in the meaning of
    - centralized control possible when needed,
    - non-variable connectivity,
    - control of the network (LAN), and high-speed
    - no firewalls, dynamic IP addresses and such restrictions

# Research Issues

- Multi-consistency
  - P2P applications to scale need to deal with various consistency models for shared state
- Algorithms
  - Dealing with completely decentralized systems
- Security, accountability
- Standards for interoperability
- Programming models, languages, platforms

# Possible Future Applications

- Effective resource utilization
  - Maybe automatic - end-user sees only illusion that he has additional CPU or disk space
  - Serious security considerations here
- Making use of peer exposure and automation
  - Exposing documents to co-workers
  - Spell-checkers making use of add-ons that my co-worker have put in (acronyms, especially).
  - Exposing address book, calendars, schedule, in different peer-groupings
  - Expose co-workers browsing (input for the intelligent assistant when I browse).

# Research at SICS

- Research on SICS on P2P systems has been going on for some time
  - until recently not called P2P
- Few examples
  - platforms/applications for P2P beyond client/server
    - DSL lab on the Mozart Programming System
    - ICE lab on distributed interactive simulation (DIVE)
  - network infrastructure for P2P
    - CNA lab on ad hoc networks

# The Future - 1

- P2P applications can be seen as a test-bed for mechanisms for
  - accountability
  - network configuration
  - experience here may impact basic services, e.g.
    - structure of the net itself (addressing)
    - basic communication protocols (TCP)
    - firewalls
    - Internet Service Providers
    - e-mail

# Future -2

- Legal restrictions on spread of digital, clonable information may in the long-term be unenforceable
  - copyright
    - make things uncloneable or at least very difficult to clone ?
    - put in public sector (royalties paid from taxes according to use estimate) ?
  - child pornography
    - resign to stopping only at the source ?
  - nazis and other extremists
    - resign to stopping only upon action ?

# Conclusion

- P2P is a class of applications that make use of potentials that client/server cannot address
  - can expect to see more (including new kinds of applications)
  - one potential is that it will be difficult/impossible to restrict information
- Demands for P2P will change the architecture of the Internet
- P2P applications are
  - extremely decentralized
  - can be extremely untrusting
  - technically challenging