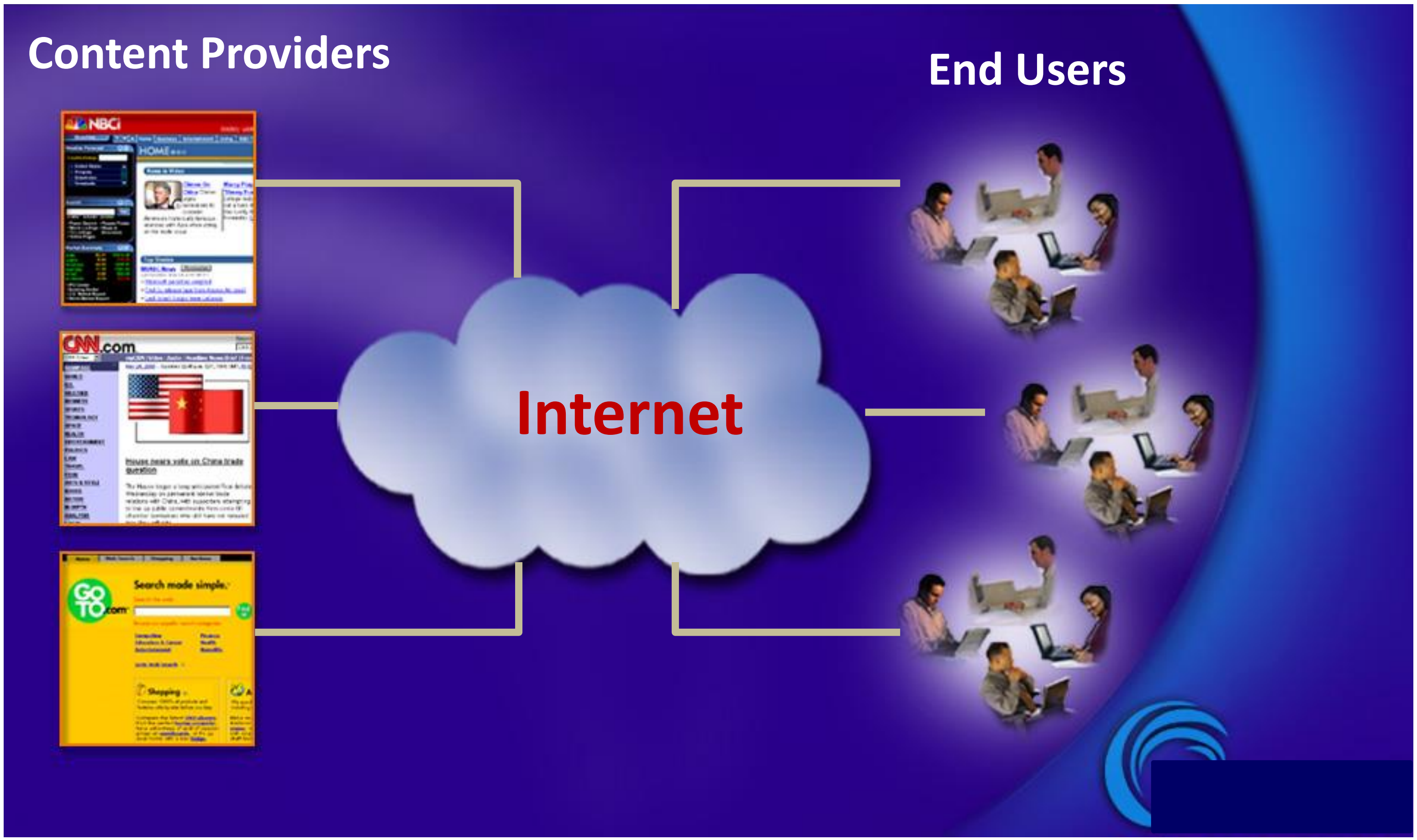
5. Content Distribution Networks and Peer-to-Peer Networks

2018 Spring

Yusung Kim
yskim525@skku.edu

Simple View for the Web Access



- Over **1.5 billion** monthly active users
- Over **1 billion hours** of video are watched every day
- **400 hours** of video are uploaded every minute
- **80%** of YouTube traffic comes from **outside** the US
- Accessed from **89 countries**

- Over **2.01 billion** monthly active users
- Photo uploads **350 million** per day
- **83%** of active users outside US & Canada
- **100 million** hours of daily video views
- **65 million** businesses have Facebook pages.
- Every 60 seconds, **3 million** items are shared.

- **Over 100 million** monthly active users
- **50%** of active users outside the US
- **1 billion hours** of video users watch per week
- Accessed from **190 countries**
- **35%** of **traffic** on North American fixed networks.

How to deliver a large amount of content to hundreds of thousands global users?

Option1: single, large "mega-server"

- Single point of failure
- Single point of network congestion
- Long path to distant clients
- Multiple copies of video sent over outgoing link

....quite simply ***not scale***

Option2: Install local proxy

- Install a local proxy in every institution
- Scalable; mitigating single point of failure.
- Who pays to install it ?
  - University? Home?, Institution?
  - KT, SKT, AT&T ?
- Difficult to estimate the proxy capacity to install
- Difficult to maintain the proxy service

….quite ***not practical***

Option3: Content Delivery Networks

- Store multiple copies of content at multiple geographically distributed sites.

- ***Type1:*** CDN servers deep into many access networks
  - used by Akamai, 1700 locations

- ***Type2:*** smaller number of larger clusters in Tier-1 ISPs
  - used by Limelight, 10's locations

- commercial CDNs
  - Akamai, Limelight, Amazon, Azure, Level3, ... etc.

Content Delivery Networks

- CDN is a **large distributed system** of servers deployed in multiple locations.

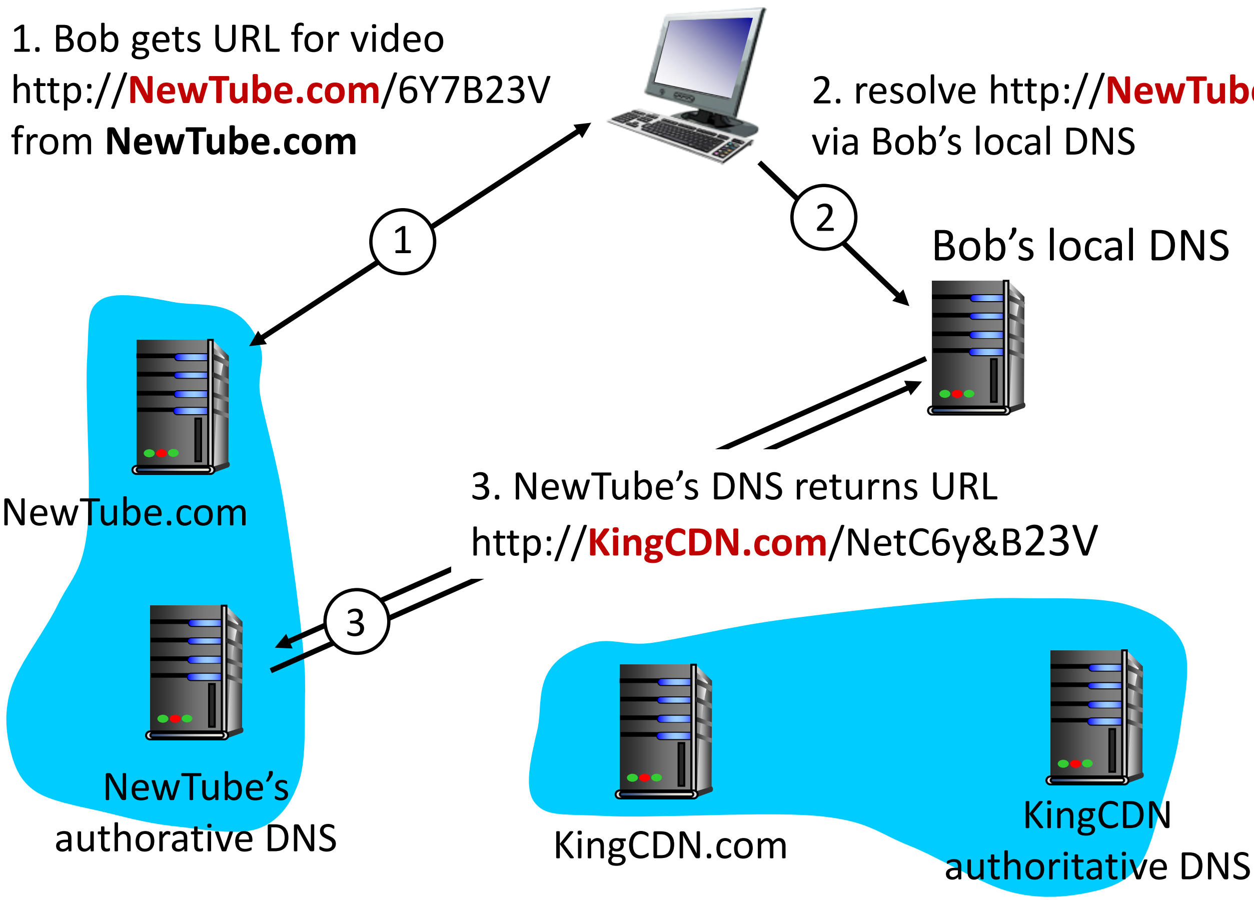
- Provide content to end-users with **high availability** and **high performance**
  - web objects (text, image, scripts)
  - files (media, software, documents)
  - media streaming (live, on-demand)
  - applications (e-commerce)

Content Delivery Networks

- **Content providers** pay **CDN** operators
- **CDN** pays **ISPs and network operators** for hosting servers
- Better user experience
- Offloading the load of content providers
- CDNs operated by content providers; Google/Microsoft
- Telecommunications CDN (Telco CDN)

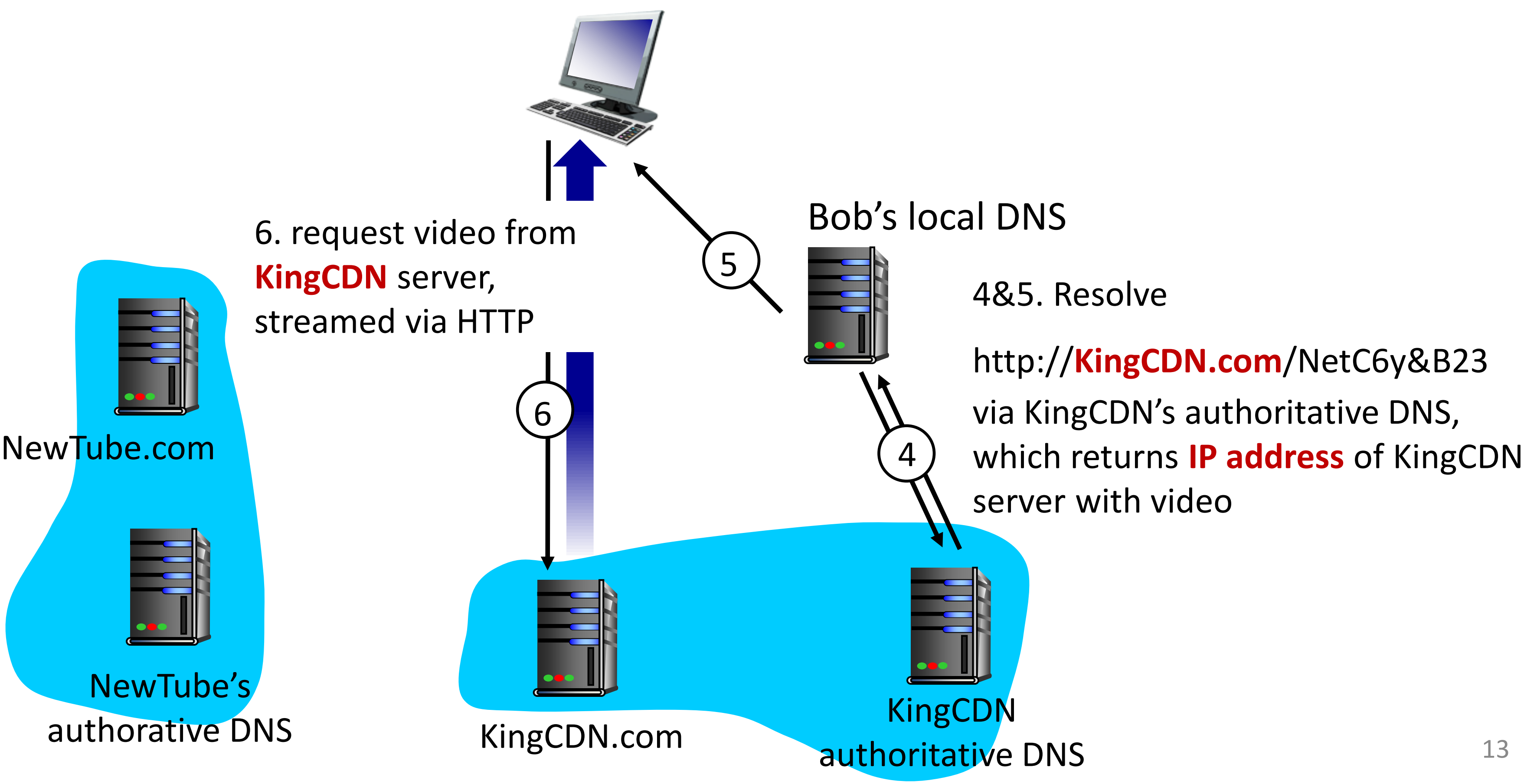
CDN: "simple" content access scenario

- Bob (client) requests video http:// **NewTube.com**/6Y7B23V
- Video is also stored in CDN at http://**KingCDN.com**/NetC6y&B23V



CDN: “simple” content access scenario

- Bob (client) requests video http://**NewTube.com**/6Y7B23V
- Video is also stored in CDN at http://**KingCDN.co**m/NetC6y&B23V



CDN server/cluster selection strategy

Challenge:

How does CDN DNS select "**good**" CDN node to stream to client ?

- Pick CDN node geographically closest to client
- Pick CDN node with shortest delay to client
  (CDN nodes periodically ping access ISPs, reporting results to CDN DNS)

Alternative:

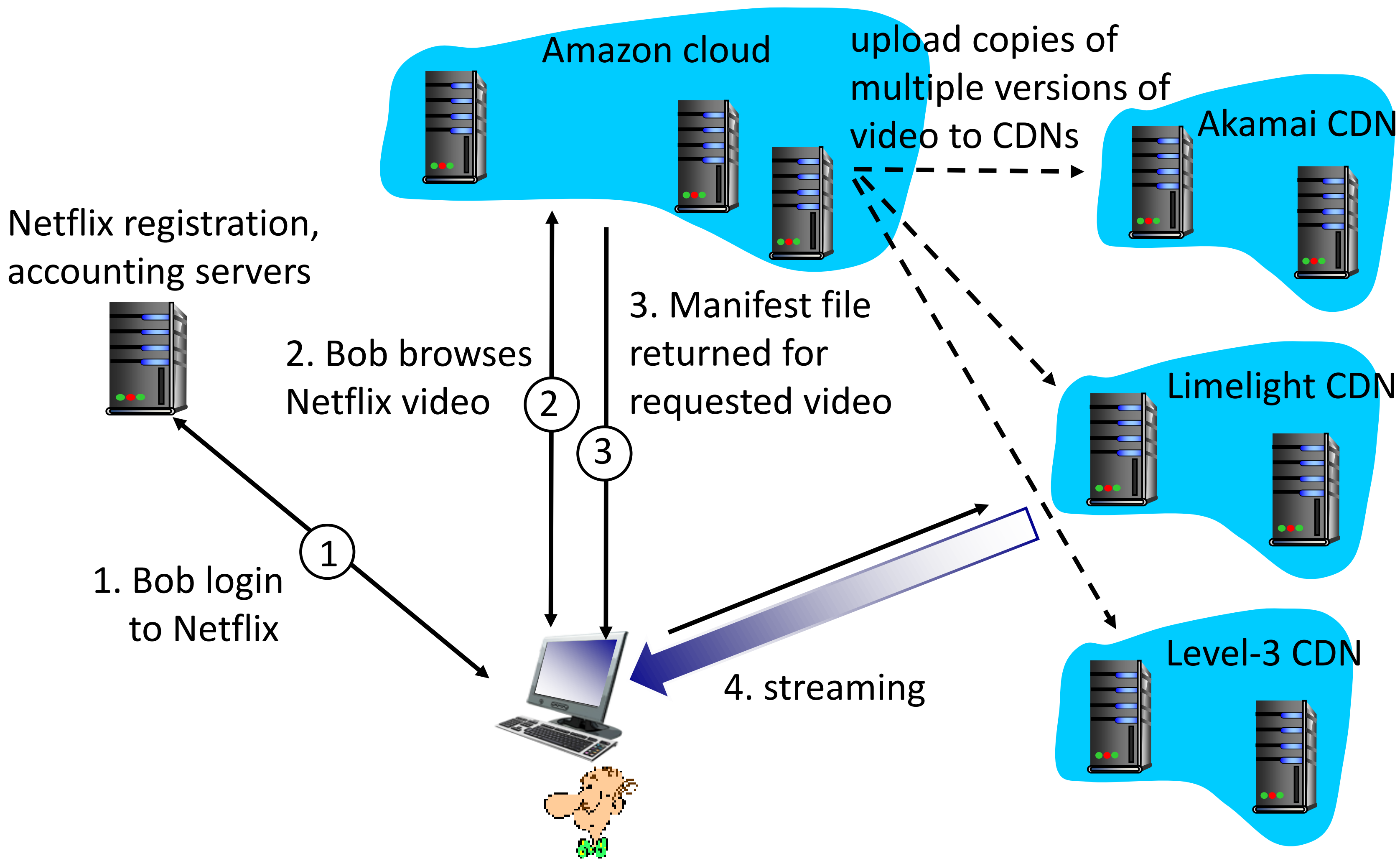
let *client* decide (give client a list of several CDN servers)

- Client pings servers, picks "best" (Netflix approach)

Case study: Netflix

- **35%** of downstream traffic in North America in 2016.
- Owns very **little infrastructure**, uses 3rd party services:
  - using Amazon cloud services
  - Netflix uploads content to Amazon cloud
  - create multiple versions (different encodings) in cloud
  - upload the versions from cloud to CDNs
  - cloud hosts Netflix web pages for user browsing
  - using three CDNs : Akamai, Limelight, Level-3

Case study: Netflix

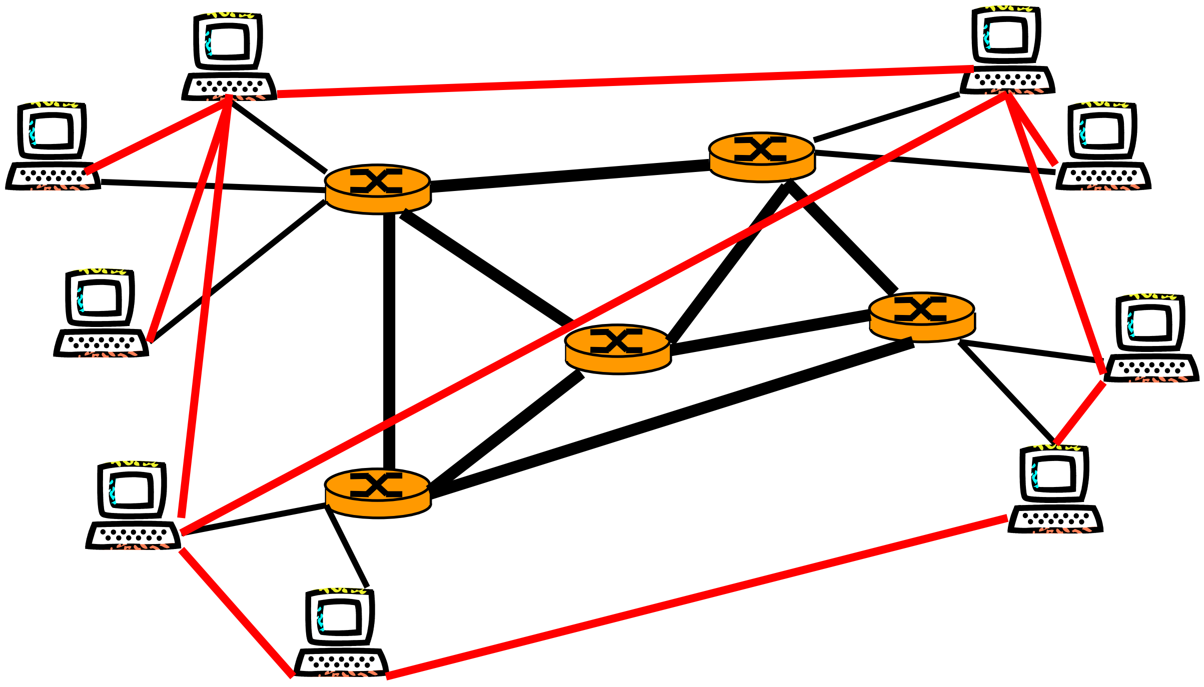


Option4: Peer-to-Peer Networks

- P2P system is a **distributed application architecture** without a central server
- Peers are **equally** privileged.
  - both client and server
- Peers **share** their resources
  - processing power, disk storage, or bandwidth
- **Overlay** construction

P2P Networks: Overlay Networks

- Overlay Network
  - Building a network at the application layer



Pure P2P architecture

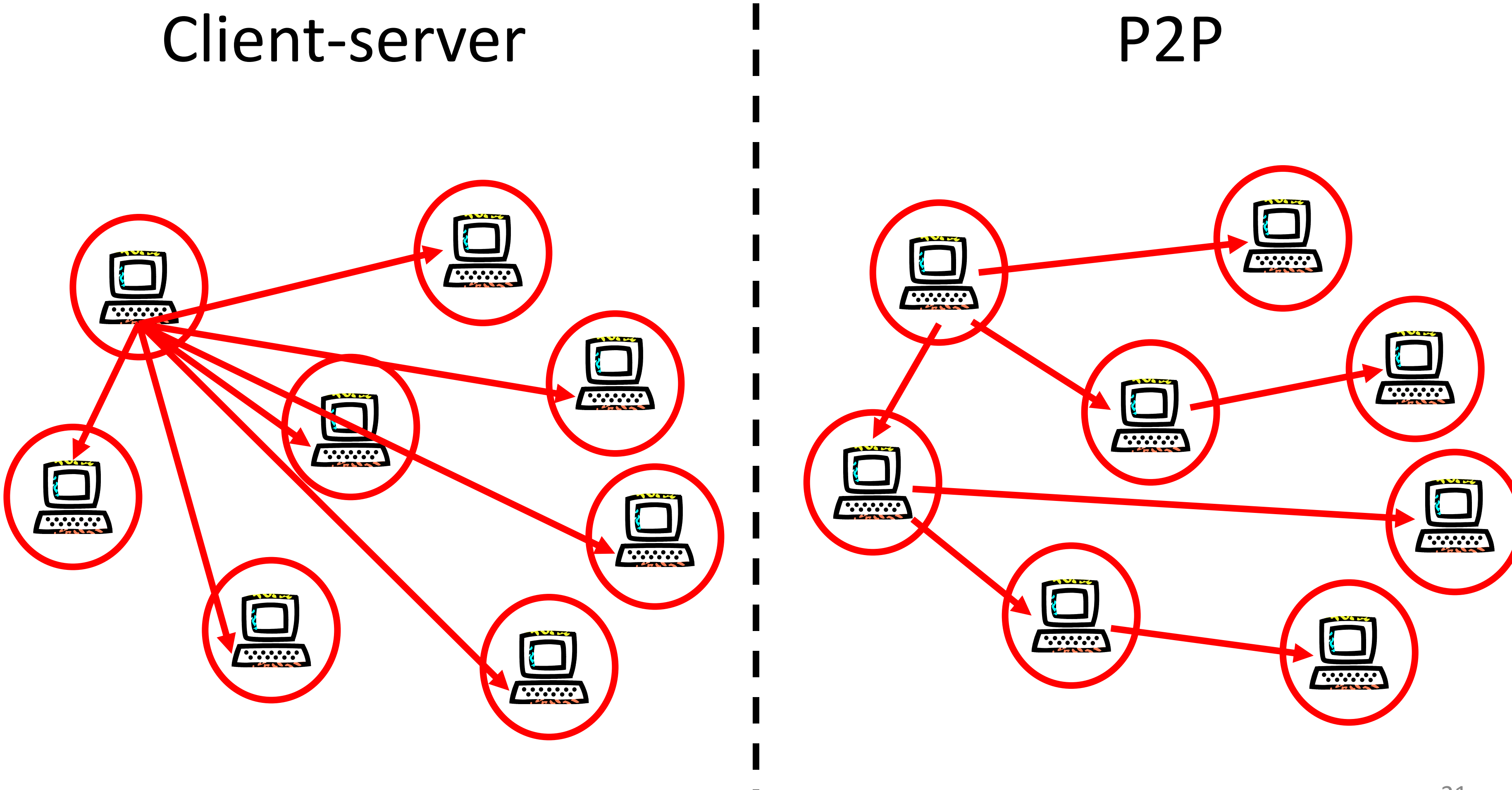
- *no* always-on server
- arbitrary end systems directly communicate
- peers are intermittently connected and change IP addresses
- Why P2P applications became popular in mid-2000 ?
  - High speed Internet connections
  - High performance of end-users
- More peers, P2P systems have greater power

What is the benefit of P2P?

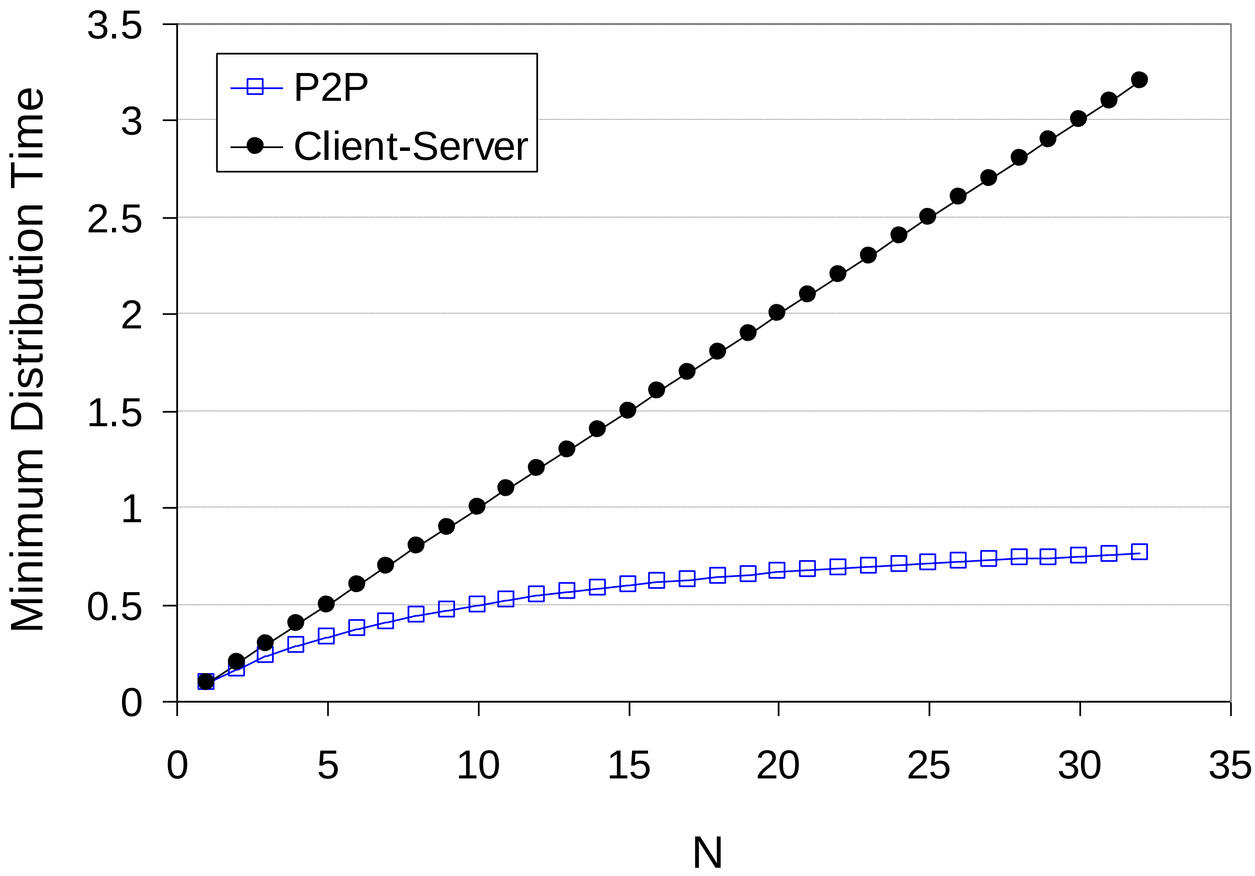
- **Scalability**
  - Peers who consume resources also donate their resources
  - Aggregate resources grow naturally with increasing the number of peers
- **Reliability**
  - Replicas
  - Geographic distribution
  - No single point of failure
- **Low cost**
  - Nodes self organize
  - No need to deploy servers to satisfy demand
  - Built-in fault tolerance, replication, and load balancing

Intuition

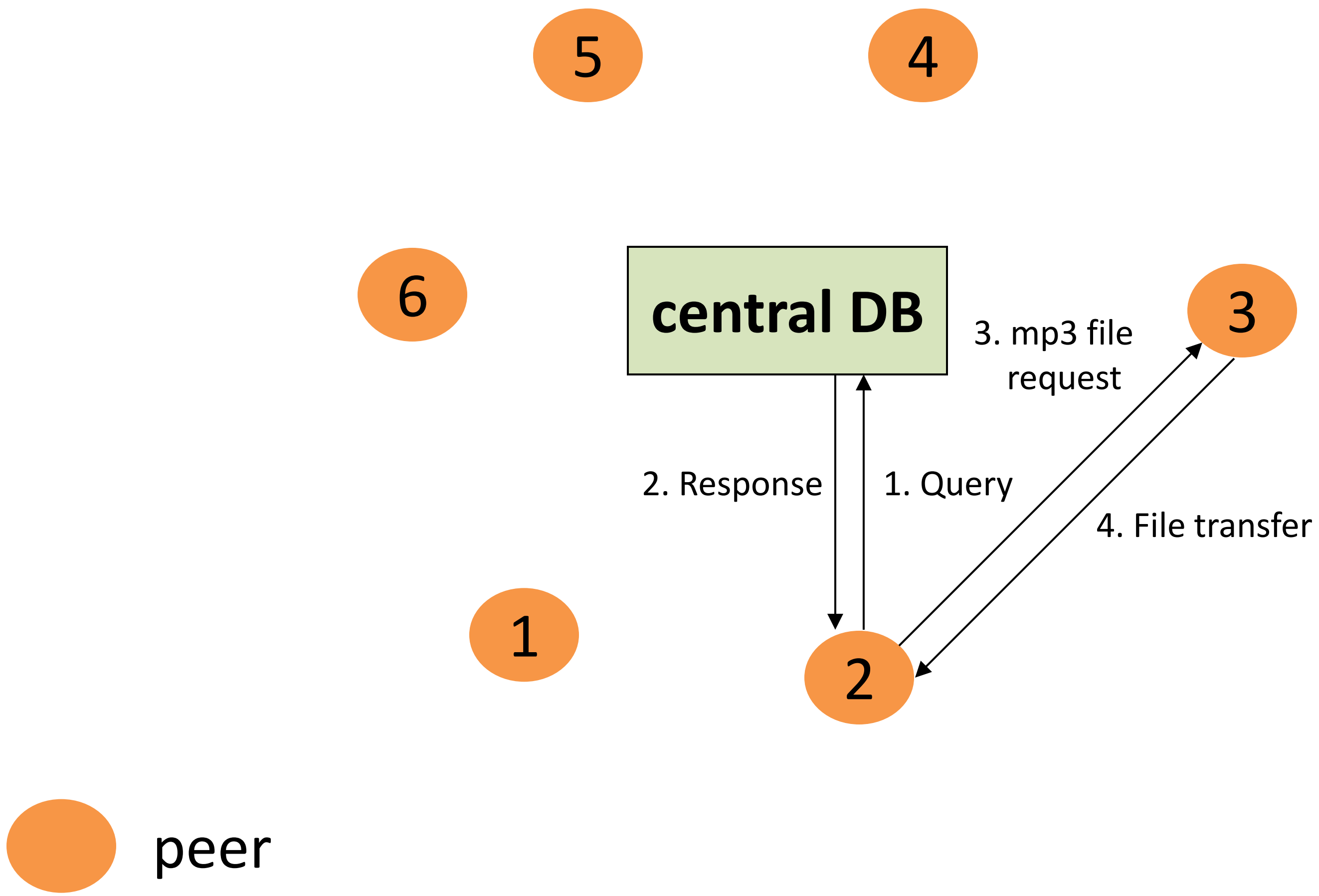
Client-server | P2P



Client-server vs. P2P: example



- **First** generation P2P for mp3 sharing
- **NOT** a pure P2P network => hybrid system
- Operations:
  - All clients send MP3 info they hold to **a server**
  - One client sends a query to the server,
    the server responds to the client with a list of clients
  - The client downloads directly from other client(s)



5
4
6
central DB
3. mp3 file request
3
2. Response
1. Query
4. File transfer
1
2
peer

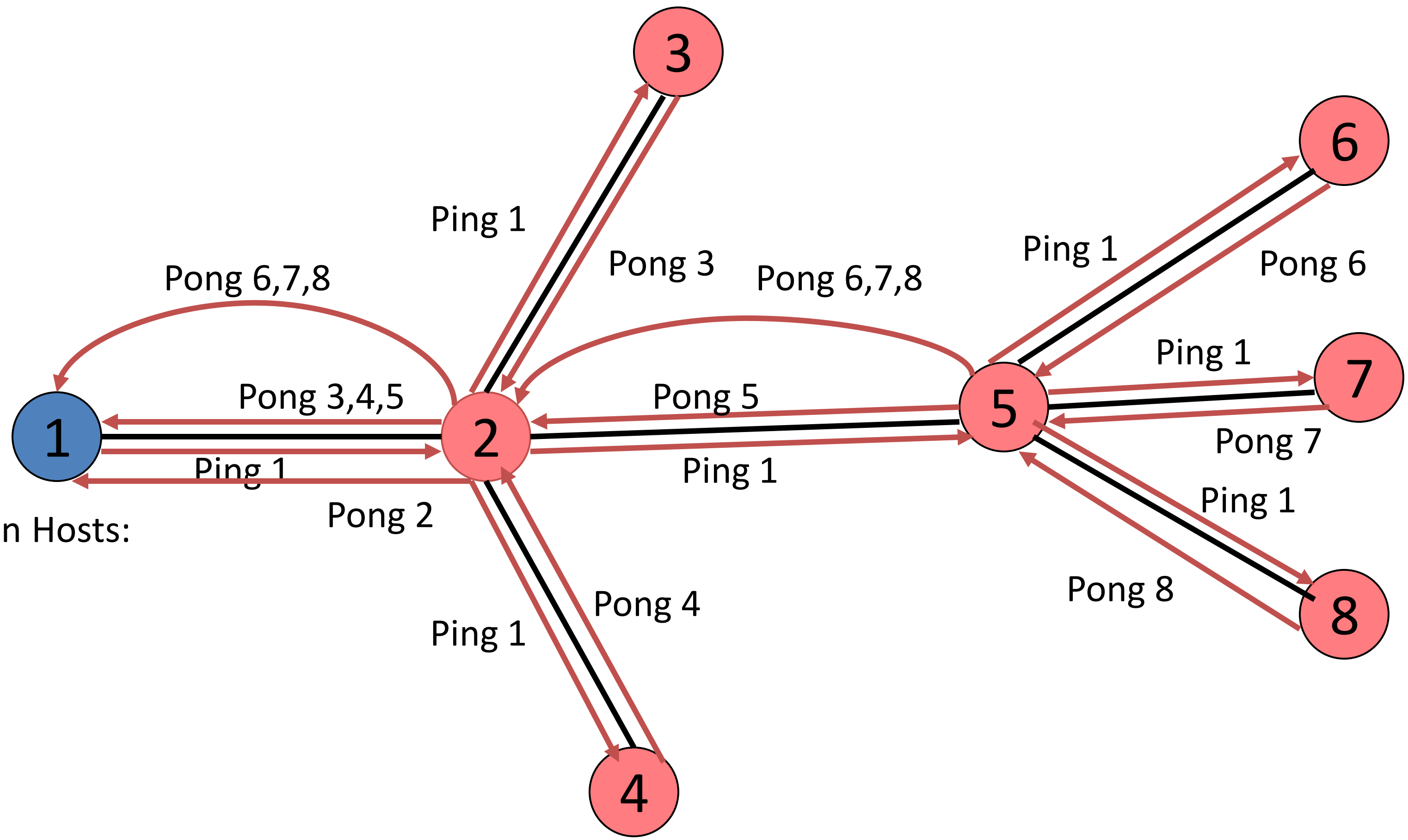
napster

- Further services:
  - Chat program, instant messaging service…
- **Query is fast**, but **centralized** system
  - Single point of failure => limited fault tolerance
  - **Legal issues**

Gnutella

- **Pure** peer-to-peer (decentralized)
- Very simple protocol
- **No** routing **"intelligence"**
- Constrained broadcast
  - Life-time of packets limited by TTL (typically set to 7)
  - Packets have unique IDs to detect loops

Gnutella - PING/PONG



Known Hosts:
2

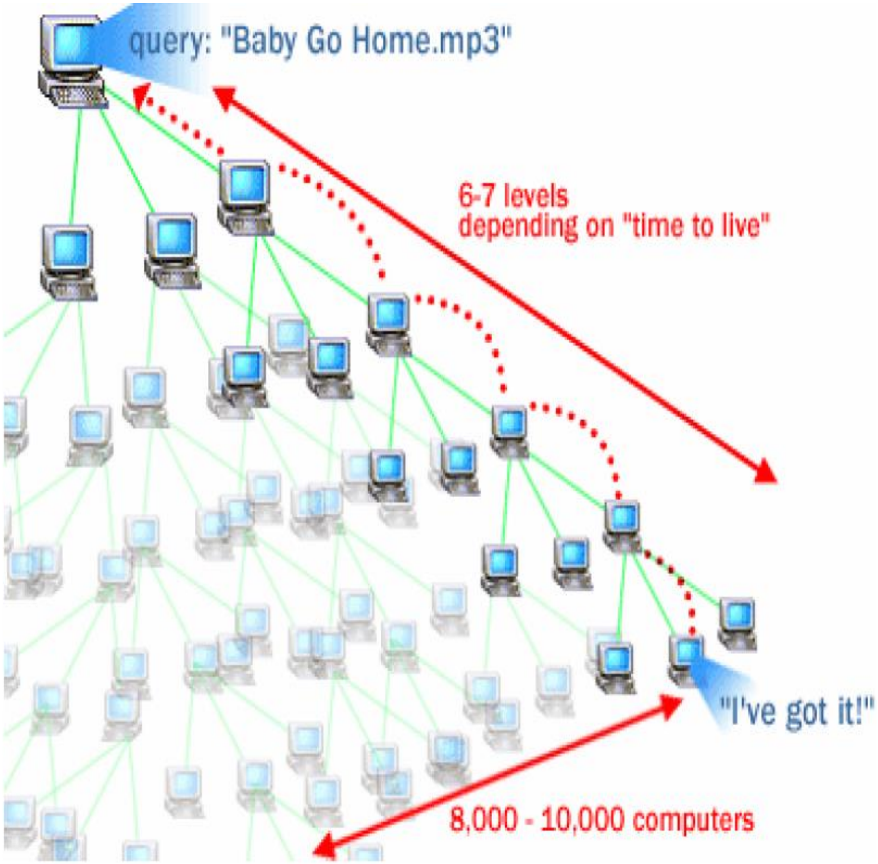
3,4,5

6,7,8

Query/Response analogous

Gnutella

- Share any type of files (not just music)
- First ask my neighbors, they ask their neighbors
- Peers with matching files reply to you
- Decentralized search
- No single point of failure
- **Flooding may be not scalable**



Structured P2P

- The second generation of P2P networks
  - Self-organizing
  - Fault-tolerant
  - **Load balanced**
- **Guarantee** small number of hops to answer query
  - Major difference with unstructured P2P systems
- Based on a distributed hash table (DHT)

Distributed Hash Table (DHT)

- DHT is a ***distributed P2P database***
  - Database has (key, value) pairs;
    e.g.   key: hash(file name); value : a node (IP address)
  - Distribute (key, value) pairs **over peers**
- A peer *queries* DHT with key, then DHT returns values that match the key
- Core operation: **find a node responsible for a key**

How to distribute (key, value) to peers

- Basic idea in **Chord [MIT 2011]**
  - Assign each key to peer by Hash(peer IP address)
  - Get a key for a file by Hash (a file name)
  - Key is ranging [0, 2^m - 1]
  - Put a file in the peer that is closest to the key.

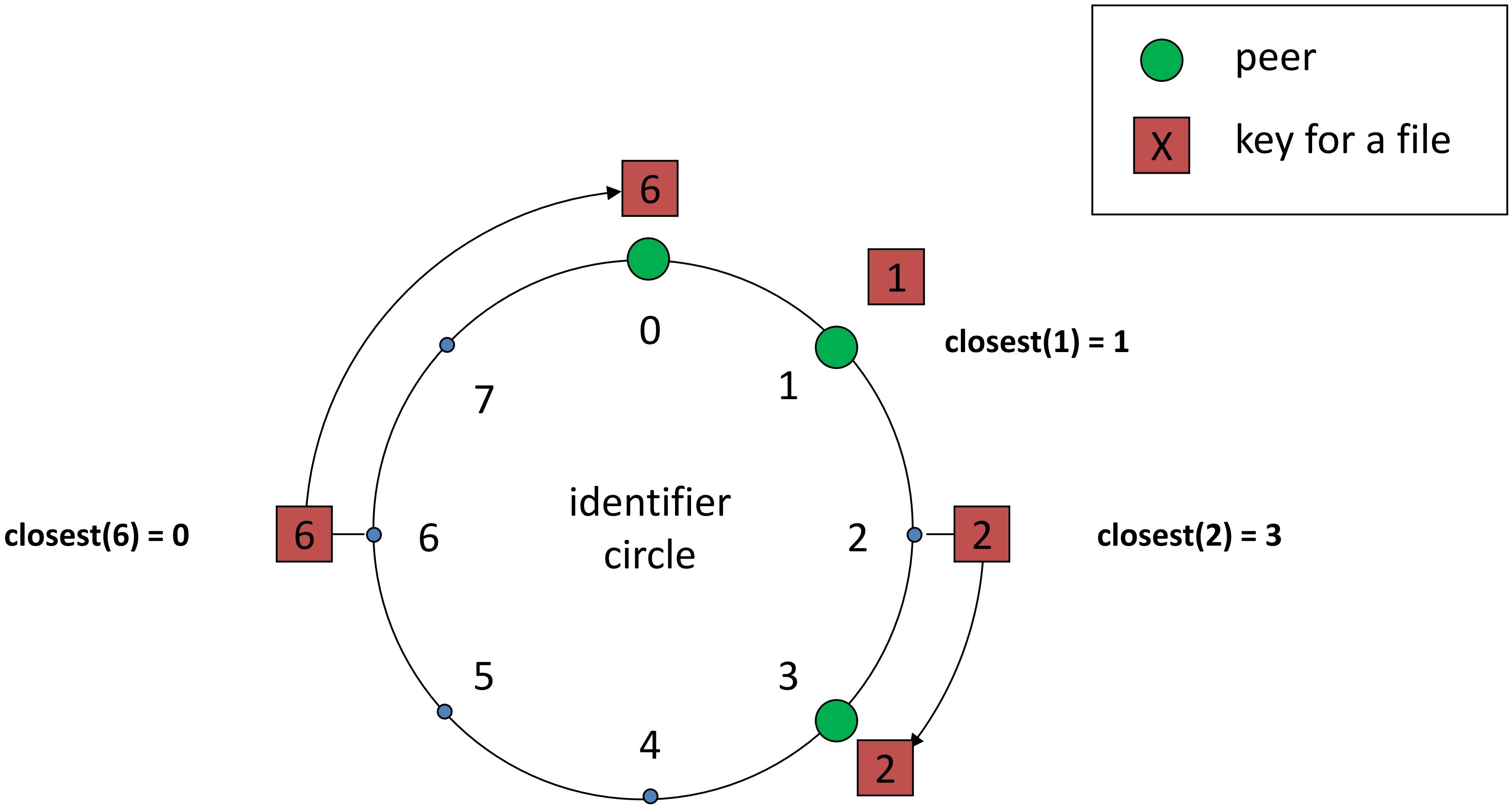
Assign keys to peers

- Assign key to the peer that has the ***closest*** ID.

- closest means the ***immediate successor*** of the key.

  e.g., *m*=4; peers: 1, 3, 4, 5, 8, 10, 12, 14;
  - For key 6, successor peer is 8
  - For key 15, successor peer is 1

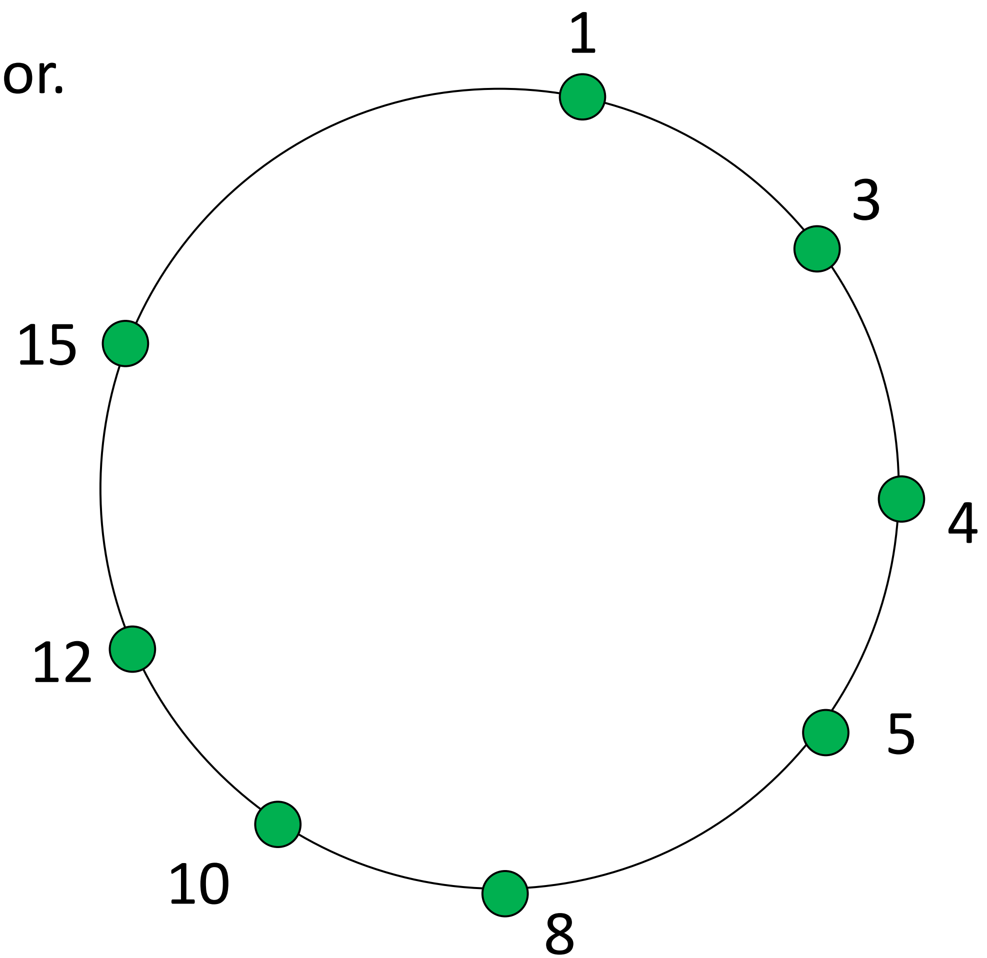
Where to upload a file



How to lookup peer for a key?

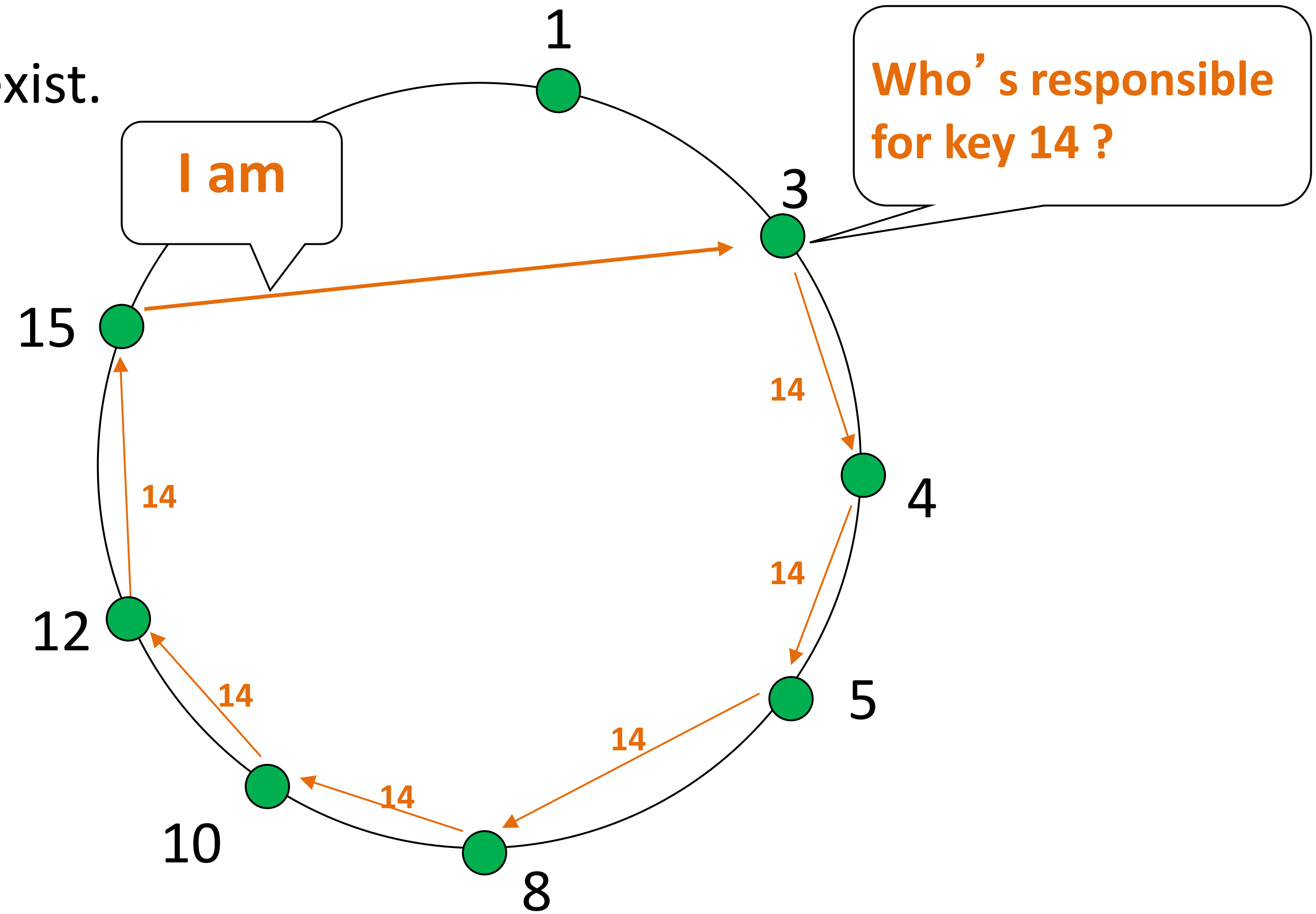
- Assume: Each peer only aware of immediate successor.

- If Key of A file is 14, how to find the closest?



How to lookup peer for a key?

O(N) messages on average to resolve query where *N* peers exist.



How to lookup peer for a key?

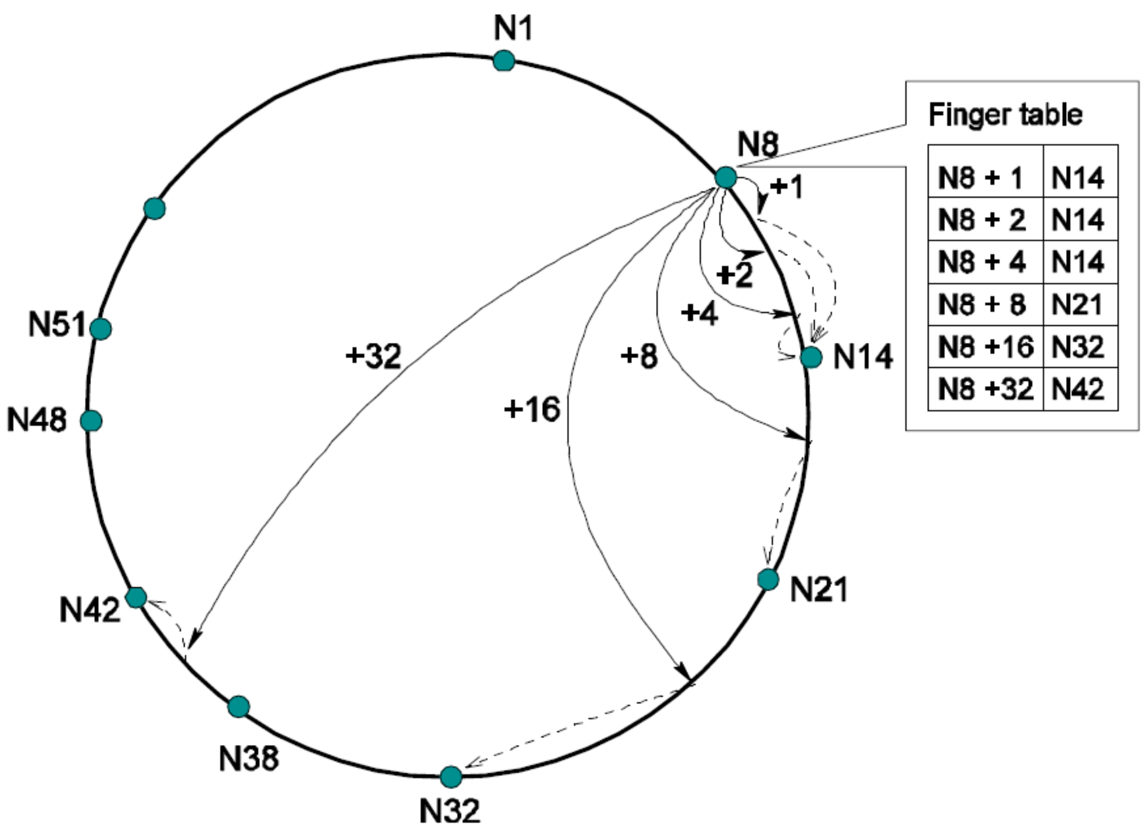
- I f each peer knows one successor IP address?
  - # queries sent is O(*N*) where *N* is the number of peers in this P2P networks.
  - maintenance cost : O(1)

- If each peer knows all peers' IP address ?
  - # queries sent is only one
  - maintenance cost : O(*N*)

A trade-off between lookup-time and maintenance-cost

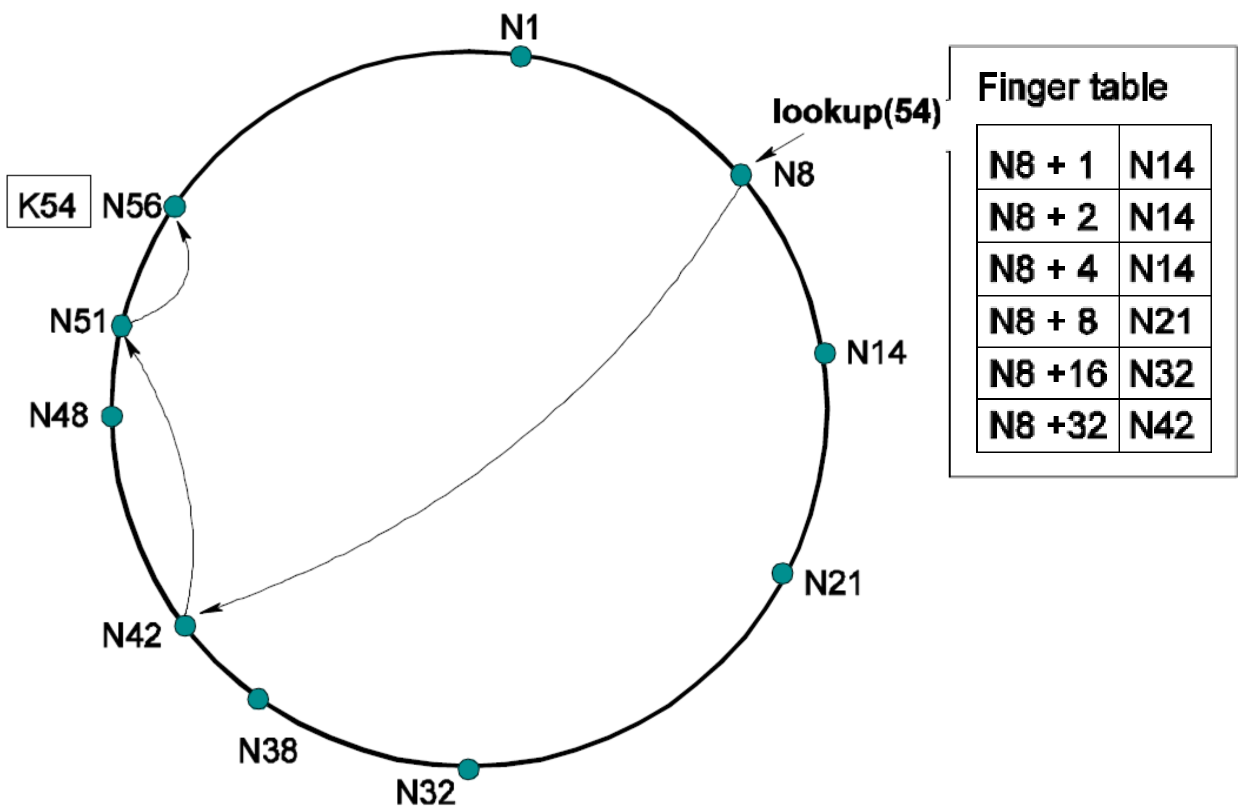
Chord Finger Table [MIT 2011]

- Finger Table size is **O(log *N*)**
- i^{th} finger points to the closest of 2^{i-1}



Chord Lookup [MIT 2011]

- Lookup the furthest node that precedes key
- Query reaches target in **O(log *N*)** hops

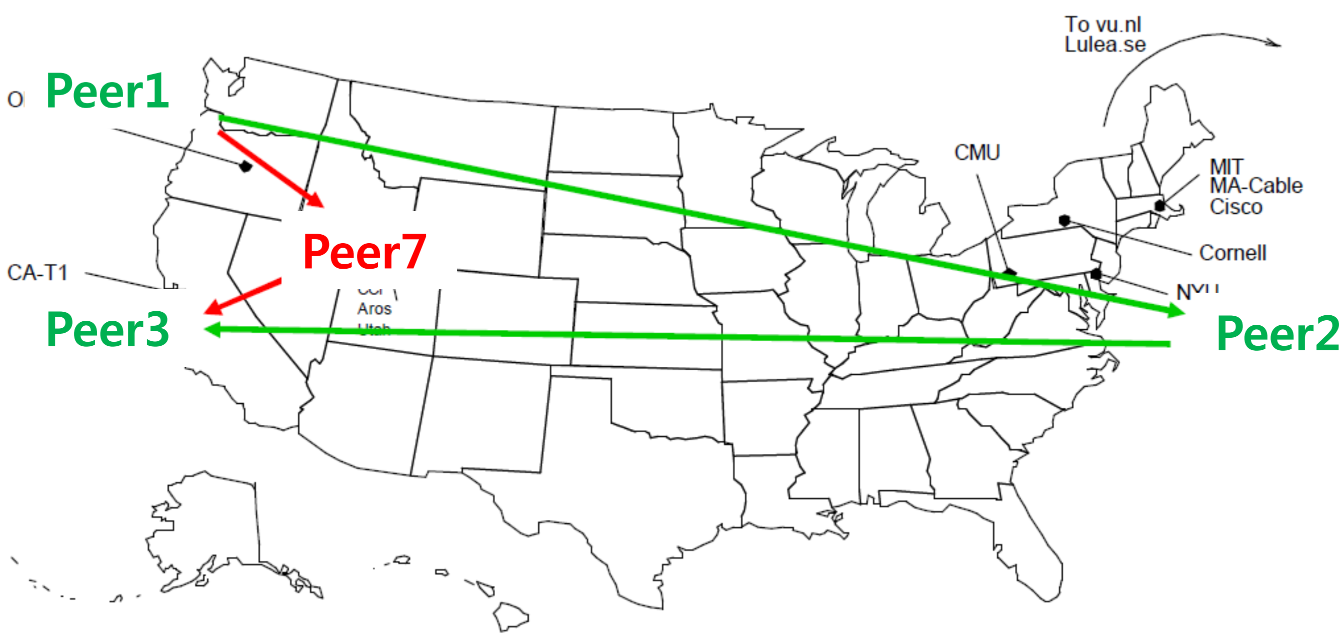


Other P2P Issues: Free riding

- P2P File sharing rely on **users sharing data**
- Free riding is that a peer uses P2P network services but doesn't contribute to the network.
- On Gnutella
  - **15%** of users contribute **94%** of content
  - **63%** of users **never responded** to a query
- How to encourage peers to keep contributing ?

Other P2P Issues: Network Locality

- Neighbor peers can be far in the physical networks.
- Peer3 may receive content from Peer1 via Peer2.
- How to construct P2P networks or provide routing topologically-aware?



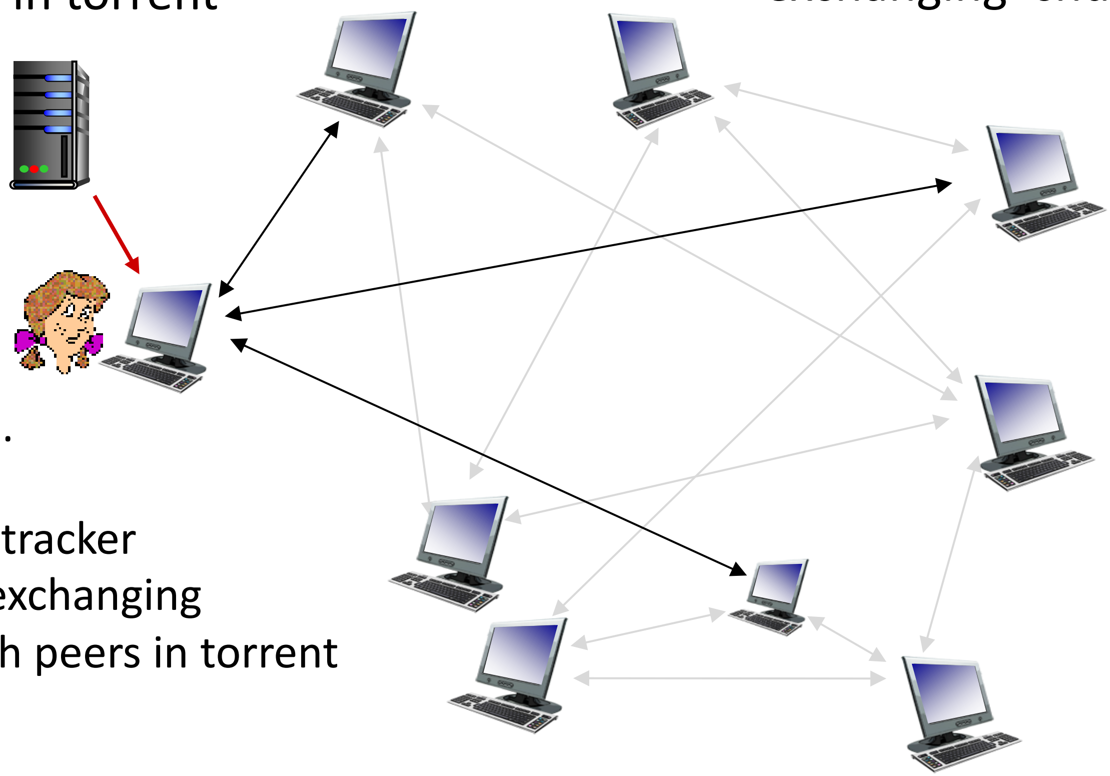
BitTorrent

BitTorrent : P2P file distribution

- File divided into 256 Kbyte chunks
- Peers in torrent send/receive file chunks

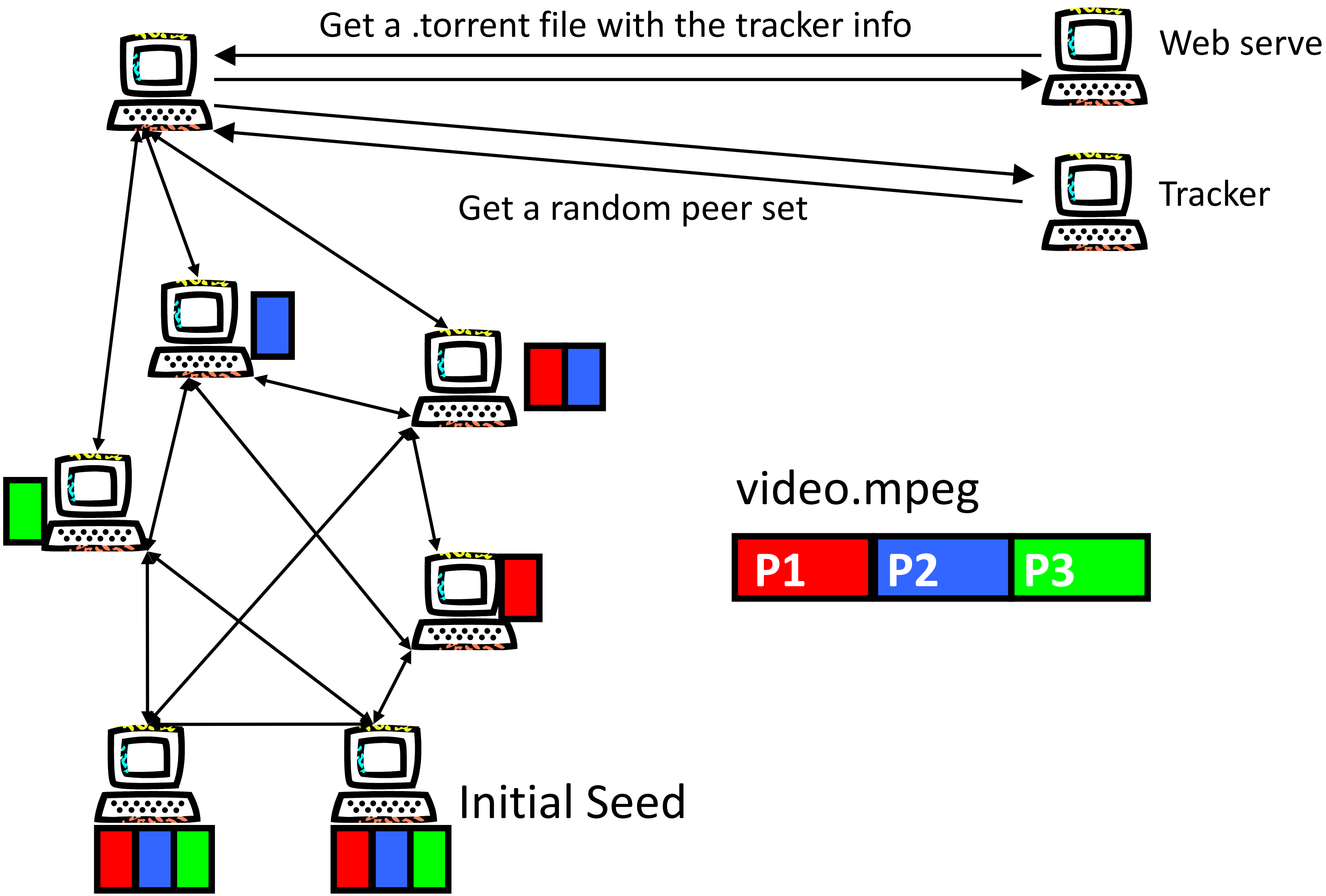
tracker: tracks peers participating in torrent

torrent: group of peers exchanging chunks of a file



Alice arrives …
… obtains list
of peers from tracker
… and begins exchanging
file chunks with peers in torrent

BitTorrent Overview



BitTorrent: requesting, sending file chunks

Requesting chunks:

- At any given time,
  different peers may have different subsets of file chunks
- Periodically,
  Alice asks each peer for list of chunks that they have
- Alice requests missing chunks from peers, rarest first

BitTorrent: requesting, sending file chunks

Sending chunks: tit-for-tat

- Alice sends chunks to top four peers currently sending her chunks *at highest rate*
  - other peers are choked by Alice (do not receive chunks from her)
  - re-evaluate top four peers every 10 seconds
- Every 30 secs: randomly select another peer, starts sending chunks
  - "optimistically unchoke" this peer (Bob)
  - Alice may became one of top four peers of Bob

higher upload rate*:* find better trading partners, get file faster !

Peak Traffic in 2011

Table 6 - Europe, Fixed Access, Peak Period, Top Applications by Bytes

	Upstream		Downstream		Aggregate	
Rank	Application	Share	Application	Share	Application	Share
1	BitTorrent	59.68%	BitTorrent	21.63%	BitTorrent	28.40%
2	Skype	7.16%	HTTP	20.47%	HTTP	18.08%
3	HTTP	7.02%	YouTube	14.13%	YouTube	11.93%
4	PPStream	3.64%	RTMP	4.58%	RTMP	3.90%
5	Spotify	2.91%	Flash Video	3.99%	Flash Video	3.38%
6	SSL	2.66%	iTunes	3.65%	SSL	3.09%
7	eDonkey	1.76%	SSL	3.18%	iTunes	3.07%
8	YouTube	1.76%	NNTP	2.73%	Skype	2.44%
9	Facebook	1.42%	Facebook	1.71%	NNTP	2.30%
10	Teredo	1.18%	Skype	1.42%	PPStream	1.77%
	Top 10	89.19%	**Top 10**	77.49%	**Top 10**	78.36%

SOURCE: SANDVINE NETWORK DEMOGRAPHICS

sandvine

Peak Traffic in Europe, 2016

	Upstream		Downstream		Aggregate	
Rank	Application	Share	Application	Share	Application	Share
1	BitTorrent	21.08%	YouTube	24.44%	YouTube	21.16%
2	HTTP	12.53%	HTTP	15.39%	HTTP	14.94%
3	YouTube	7.51%	Facebook	7.56%	BitTorrent	8.44%
4	SSL - OTHER	7.43%	BitTorrent	6.07%	Facebook	7.39%
5	Facebook	6.49%	SSL - OTHER	5.51%	SSL - OTHER	5.81%
6	Skype	4.78%	Netflix	4.82%	Netflix	4.18%
7	eDonkey	3.67%	MPEG - OTHER	3.82%	MPEG - OTHER	3.51%
8	MPEG - OTHER	1.89%	iTunes	2.24%	iTunes	2.03%
9	Apple iMessage	1.70%	Flash Video	1.85%	Skype	1.78%
10	Dropbox	1.44%	Twitch	1.65%	Flash Video	1.59%
		68.54%		73.35%		70.84%

sandvine

Table 1 - Top 10 Peak Period Applications - Europe, Fixed Access

Peak Traffic in Asia, 2016

	Upstream		Downstream		Aggregate	
Rank	Application	Share	Application	Share	Application	Share
1	BitTorrent	48.22%	YouTube	29.31%	BitTorrent	24.95%
2	QVoD	8.89%	BitTorrent	19.20%	YouTube	24.64%
3	Thunder	3.91%	HTTP	9.65%	HTTP	8.39%
4	HTTP	3.29%	Facebook	3.65%	Facebook	3.27%
5	Skype	2.10%	MPEG - OTHER	3.11%	Thunder	2.32%
6	Facebook	1.71%	Thunder	1.93%	QVoD	2.31%
7	SSL - OTHER	1.21%	SSL - OTHER	1.66%	SSL - OTHER	1.57%
8	PPStream	0.81%	Flash Video	1.21%	iTunes	1.26%
9	Dropbox	0.70%	Valve's Steam Service	1.16%	Skype	1.12%
10	Apple iMessage	0.57%	Dailymotion	0.88%	Flash Video	1.09%
		71.41%		71.76%		70.92%

sandvine

Table 3 - Top 10 Peak Period Applications - Asia-Pacific, Fixed Access

Peak Traffic in North America, 2016

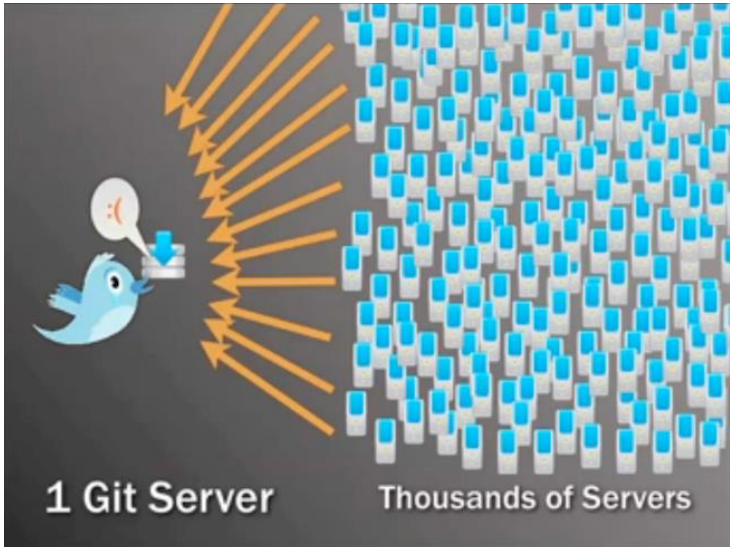
Upstream		Downstream		Aggregate	
BitTorrent	18.37%	Netflix	35.15%	Netflix	32.72%
YouTube	13.13%	YouTube	17.53%	YouTube	17.31%
Netflix	10.33%	Amazon Video	4.26%	HTTP - OTHER	4.14%
SSL - OTHER	8.55%	HTTP - OTHER	4.19%	Amazon Video	3.96%
Google Cloud	6.98%	iTunes	2.91%	SSL - OTHER	3.12%
iCloud	5.98%	Hulu	2.68%	BitTorrent	2.85%
HTTP - OTHER	3.70%	SSL - OTHER	2.53%	iTunes	2.67%
Facebook	3.04%	Xbox One Games Download	2.18%	Hulu	2.47%
FaceTime	2.50%	Facebook	1.89%	Xbox One Games Download	2.15%
Skype	1.75%	BitTorrent	1.73%	Facebook	2.01%
	69.32%		74.33%		72.72%

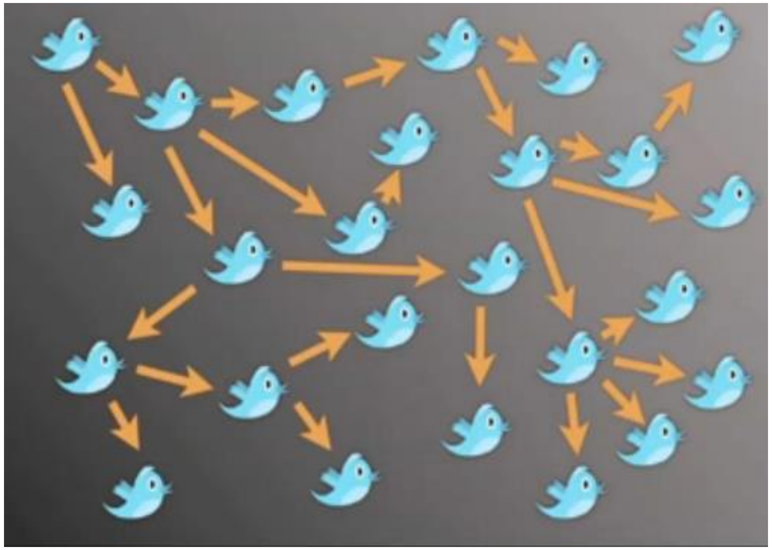
sandvine

Table 1 - Top 10 Peak Period Applications - North America, Fixed Access

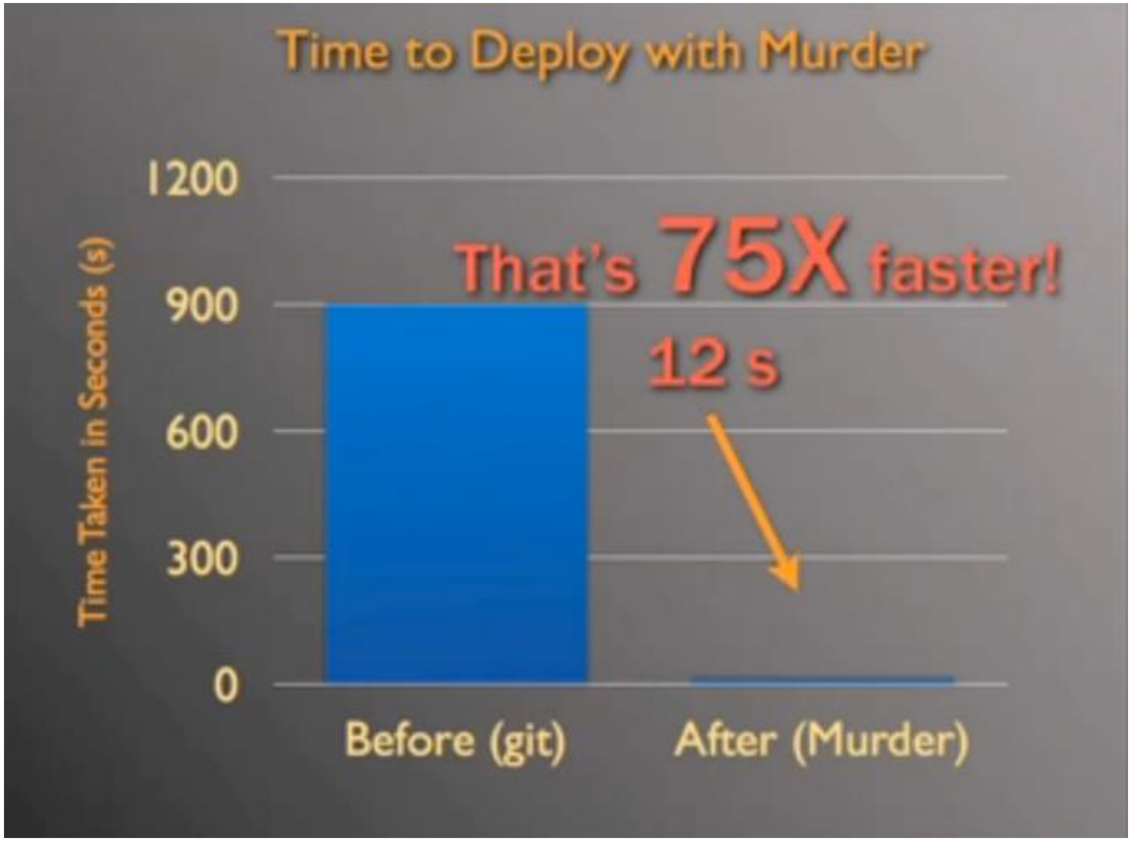
Different Usages of BitTorrent

- BitTorrent is very good to distribute content
  - Start to be used by several big companies
- Twitter is using Murder to update Twitter servers





Murder Performance



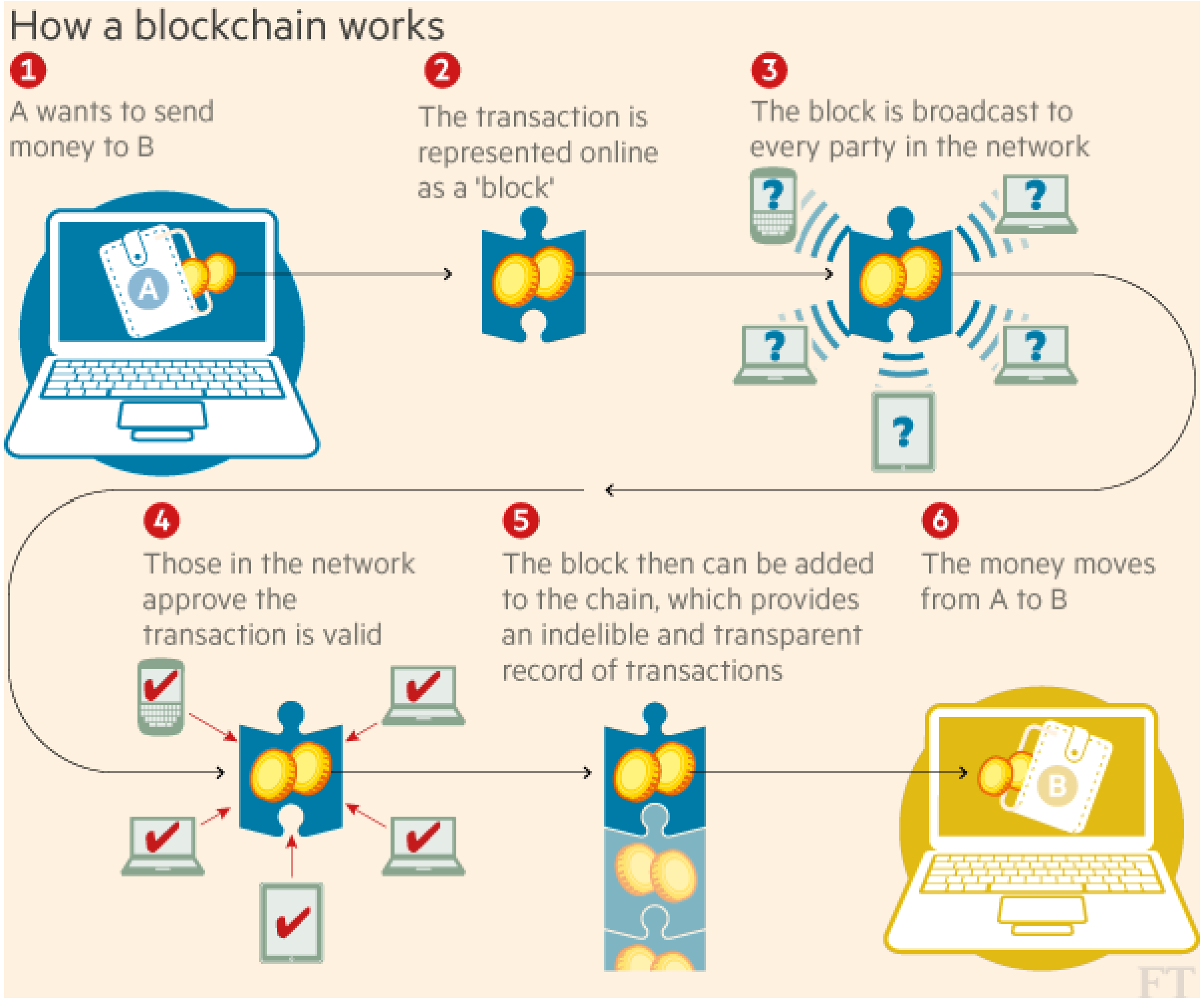
- **Bitcoin** is a worldwide cryptocurrency and the first decentralized digital payment system.

- The system is **peer-to-peer**, and transactions take place between users directly.

- These transactions are verified by peers and recorded in a public distributed ledger called a blockchain.

- There are 3 to 6 million unique users in 2017.

How a Blockchain Works



Blockchain Use Cases: Comprehensive Analysis & Startups Involved

Blockchain Startup Landscape

Blockchain Consulting/App Dev
ConsenSys, Zerado, BTL, LedgerLabs, Vanbex, appliedblockchain

Payments
Circle, bitpay, ABRA, CoinPip, Align Commerce, BitPesa, vogogo

Identity & Reputation
NETKI, Cambridge Blockchain, uport, ShoCard, civic, BLOCKSCORE

Governance & Transparency
OTONOMOS, BoardRoom, BITNATION, BALANCE

Mining
BITMAIN, BitFury, Genesis Mining, 21

Exchange, Trading & Investing
coinbase, GEMINI, Mirror, BITSTAMP, HUOBI.com, BTCC, LedgerX, kraken, coinfloor, POLONIEX, bitFlyer, WAVES, SHAPESHIFT, OKCoin, BITFINEX

Media
coindesk, CryptoCompare, BTCMEDIA, BRAVE NEWCOIN., COINTELEGRAPH

Legal, Audit & Tax
PEERNOVA, Libra, clause

Content Management
ALEXANDRIA, LBRY, Revelator, OD JAAK, ujo MUSIC, DECENT, SINGULAR, res( )nate

Data Analytics, Compliance & Security
elliptic, simplex, Skry, CHAINALYSIS, coinfirm, airBitz

Social Network
synereo, AKASHA, steemit

Wallet
xapo, Jaxx, BLOCKCHAIN, TREZOR, CiPHREX, BITX, Ledger Wallet, colu.

Data Provenance & Notary
factom, PROVENANCE, bitproof.io, guardtime, blockai, BLOCKVERIFY, everledger, TIERION

Supply Chain & Logistics
FILAMENT, Hijro, Blockfreight, skuchain

@vijaymichalik @Frost_Sullivan @lawrencelundy @OVioHQ

Prediction Markets
GNOSIS, augur, Hivemind

Public Chain Infrastructure
Bitcoin Foundation, ZCASH, MONERO, Blockstream, ethereum, Lisk, parity, blockstack, INFURA

Commerce & Advertising
OB1, Purse, safemarket, brave

Financial Services Infrastructure
ripple, Digital Asset Holdings, PAXOS, R3, MELONPORT, clearmatics, TALLYSTICKS, UBIKI, SETL, symbiont, t0, Chain, AXONI

Enterprise Infrastructure
Gem, Enterprise Ethereum Alliance, CREDITS, bloq, BIGCHAINDB, Filecoin, STORJ, nuco, Tendermint, HYPERLEDGER PROJECT, MONAX

FROST & SULLIVAN   OUTLIER•VENTURES