Distributed Systems

Today

- Peer-to-peer systems
- Next Tues., no class.
- Next Thurs., distributed shared memory.

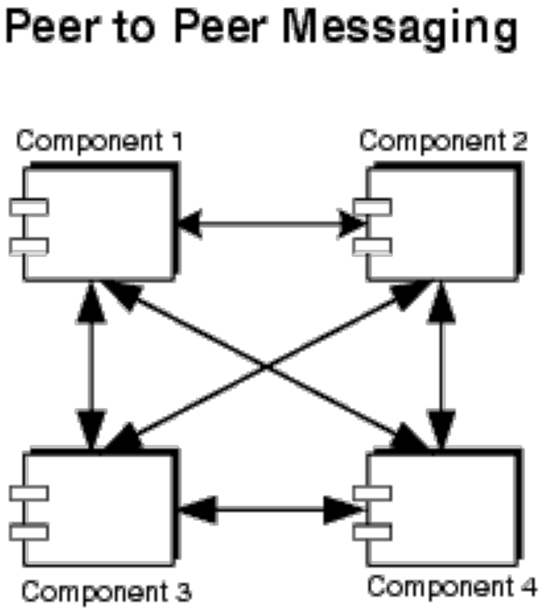
Peer-to-peer

- What distinguishes peer-to-peer from some other model?

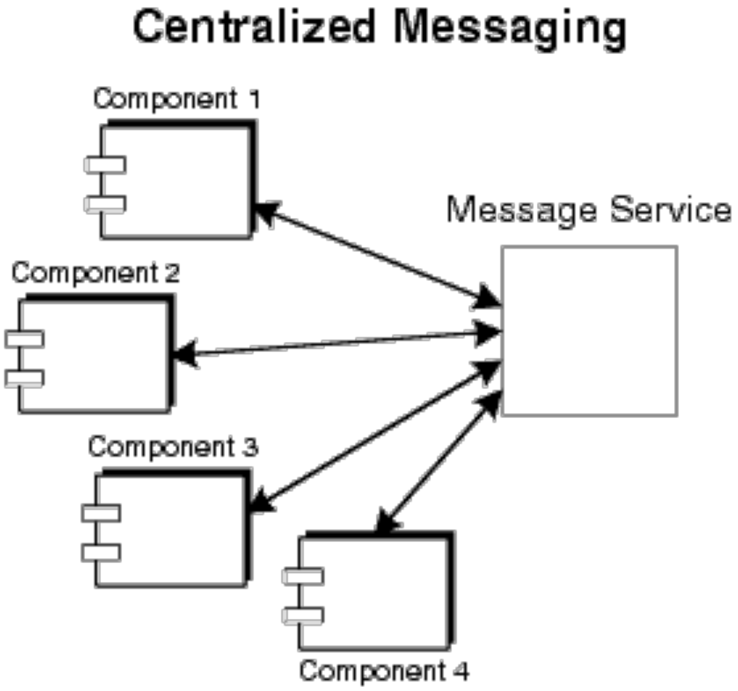
- Decentralized. All participants provide same capabilities and have the same responsibilities.
  - May not have same resources though.
  - That doesn't matter – symmetry is in the interaction and potential capability, not the actual runtime capability.

- Users contribute resources to the system.
  - Instead of resources coming from the core of the network (as in a client/server system), resources come from the edges of the network.

Peer-to-peer vs. client/server



Peer to Peer Messaging



Centralized Messaging

http://docs.sun.com/source/819-0069/concepts.html

Consequences

- Anonymity can be achieved.
  - No single well-known server.
  - Client interactions are with peers, likely anonymized through the use of unique IDs instead of IPs.
  - Tor/Vidalia are an example of a P2P-style system with the sole purpose of achieving anonymity.

- Efficiency and effectiveness of P2P system is dependent on how data or work is distributed amongst the peers.
  - No resource is guaranteed to be highly available.
  - Mechanisms to deal with unreliable participants (e.g.: replication) have a benefit that detection of byzantine failures and malicious participants is possible.

Evolution

- Various experimental, small scale projects for many years.
  - Consider server-server interactions, where servers become clients.
    - Like DNS.
  - This has a peer to peer flavor to it.

- First generation widespread P2P: Napster
- Second generation: Bittorrent, Gnutella, etc…
- Current: General purpose frameworks.

Peer-to-peer networking

- One of the interesting aspects of peer to peer systems is that they can use their own routing scheme layered over IP.
  - “Routing overlay”

Overlay routing vs IP

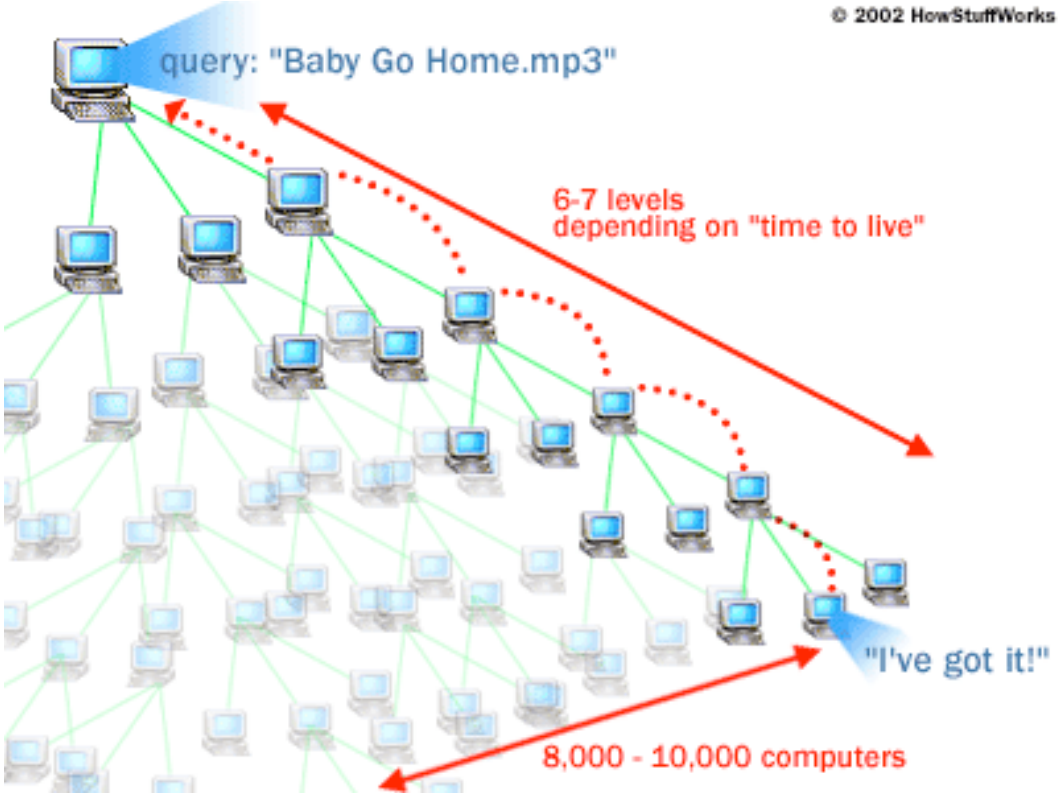
	IP	*Application-level routing overlay*
Scale	IPv4 is limited to 232 addressable nodes. The IPv6 name space is much more generous (2128), but addresses in both versions are hierarchically structured and much of the space is pre-allocated according to administrative requirements.	Peer-to-peer systems can address more objects. The GUID name space is very large and flat (>2128), allowing it to be much more fully occupied.
Load balancing	Loads on routers are determined by network topology and associated traffic patterns.	Object locations can be randomized and hence traffic patterns are divorced from the network topology.
Network dynamics (addition/deletion of objects/nodes)	IP routing tables are updated asynchronously on a best-efforts basis with time constants on the order of 1 hour.	Routing tables can be updated synchronously or asynchronously with fractions of a second delays.
Fault tolerance	Redundancy is designed into the IP network by its managers, ensuring tolerance of a single router or network connectivity failure. *n*-fold replication is costly.	Routes and object references can be replicated *n*-fold, ensuring tolerance of *n* failures of nodes or connections.
Target identification	Each IP address maps to exactly one target node.	Messages can be routed to the nearest replica of a target object.
Security and anonymity	Addressing is only secure when all nodes are trusted. Anonymity for the owners of addresses is not achievable.	Security can be achieved even in environments with limited trust. A limited degree of anonymity can be provided.

In the beginning there was…

- Napster
- Music file sharing.
- Centralized indexes, but actual files resided on user computers.
- User client sent file information to index server.
  - Peers connect to server to find a set of users that provide a song.
  - Peers then connect to user client to download song.
  - Single peer may be connected to many others downloading many songs at once.
- Interesting from a CS standpoint – first large scale peer to peer system.
  - Sketchy from the legal standpoint though…

Gnutella/Freenet

- Second-generation peer to peer systems.
- Similar to Napster, but based on decentralized index.



Decentralized approach

- Positive
  - No single point of failure.
    - Also no single point where legal orders can shut down the system.
  - Adapts to failures very easily.
- Negative
  - Takes time to execute queries.
  - User clients are part of both the data exchange (you provide data) and lookup (answering/forwarding queries) process, so some bandwidth expended to participate in the system, even passively.

Applications

- Peer-to-peer isn't just about sharing copyrighted files illegally.
- Legitimate purposes:
  - Anonymizing traffic
  - Distributed computing
  - Efficient file distribution
  - Robust data storage
  - …and anything else that can fit outside the strict client/server model.
    - Any system that has symmetry where clients can be servers and vice versa qualifies to be a form of peer to peer system.

Consistent theme

- Every user in the network has some amount of resources available to them.
  - Bandwidth
  - CPU time
  - Memory/hard disk
- Peer to peer systems most often attempt to use free resources that a user does not need.
  - Filesharing: bandwidth and hard disk space
  - Compute services: CPU time, memory

Requirements

- Scalability
  - Goal is to allow thousands or millions of users.
- Load balancing
  - System adapts based on changing loads and membership.
- Optimize for locality.
  - Place resources near others that access them.
- Dynamic host availability.
  - Plan for frequent host turnover and attempt to make it transparent while maintaining high quality of service.

Requirements (2)

- Security
  - Protect data that is distributed to avoid/detect tampering and corruption (both malicious and benign).
- Anonymity
  - Resist censorship, give data holding clients plausible deniability of responsibility for holding it.

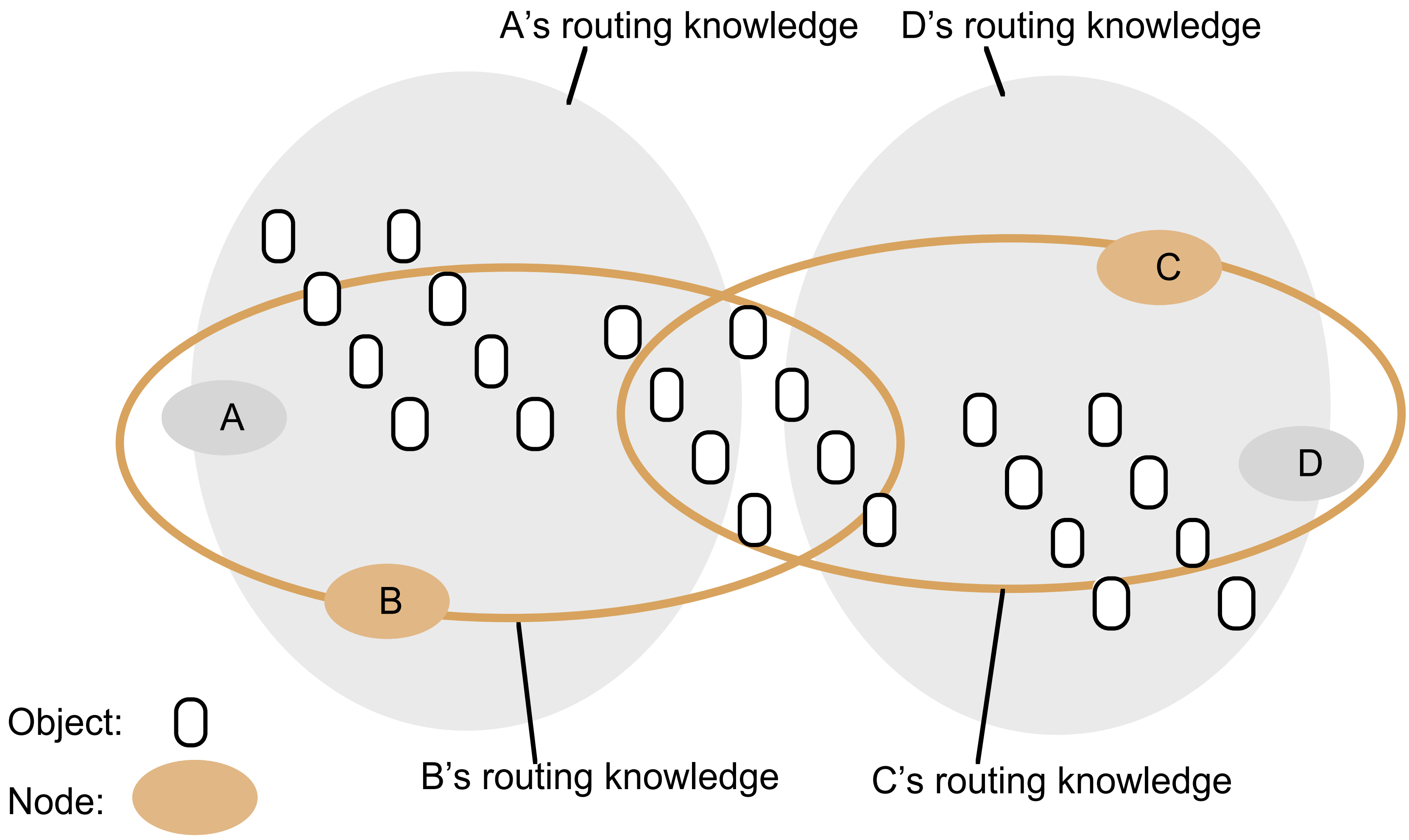
Routing overlays

- Layer of middleware responsible for locating nodes and objects.
- Layer handles routing requests to nodes holding objects.
  - Deals with nodes and objects coming and going.

- Globally unique identifiers (GUIDs) used to name hosts and objects in the system.
  - GUIDs are opaque identifiers : they reveal nothing about the location of the objects.

- Objects are replicated in the system. Routing overlay simply must find an instance of the object matching the GUID, preferably near the requestor.

Information in a routing overlay



GUIDs

- Typically hashes.
  - E.g.: SHA-1
- Provides a unique identifier for an object.
- In the simplest form, the peer to peer system supports get/put operations:
  - Get(GUID)
  - Put(GUID,data)
  - Remove(GUID)
- This sometimes called a *distributed hash table*.

Distributed hash table

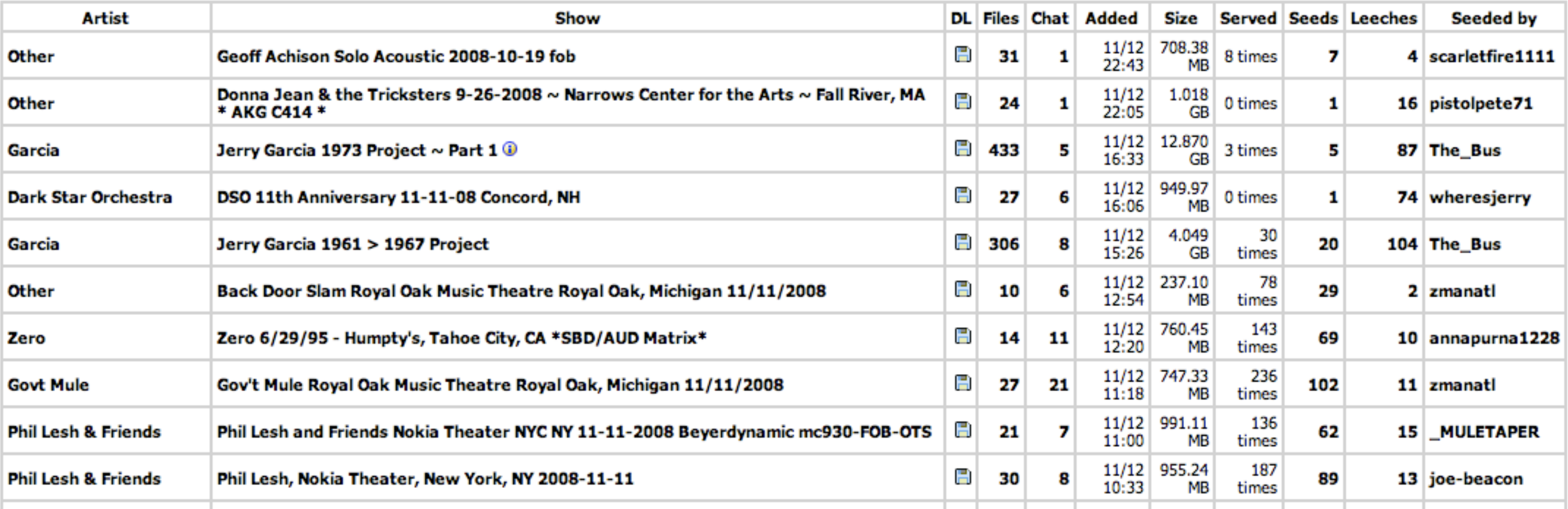
- DHT provides clients a simple put/get abstraction.
- Each peer in the system holds a subset of the GUIDs in the whole system.
- A put operation hands the data and GUID to the peer X with the closest GUID (assume peers themselves have GUIDs), and the set of r peers with GUIDs closest to that of X.

Routing

- Routing within the routing overlay is an active area of research.
- A simple method is to use *prefix routing*, which uses increasingly long sequences of digits from the GUIDs to narrow down the location of an object from coarse grained to finer until the object is found.
- This is essentially embedding the objects within the system in a 1D space. Research has been performed using higher dimensional embeddings to help narrow the search during hops through the overlay.

Bootstrapping

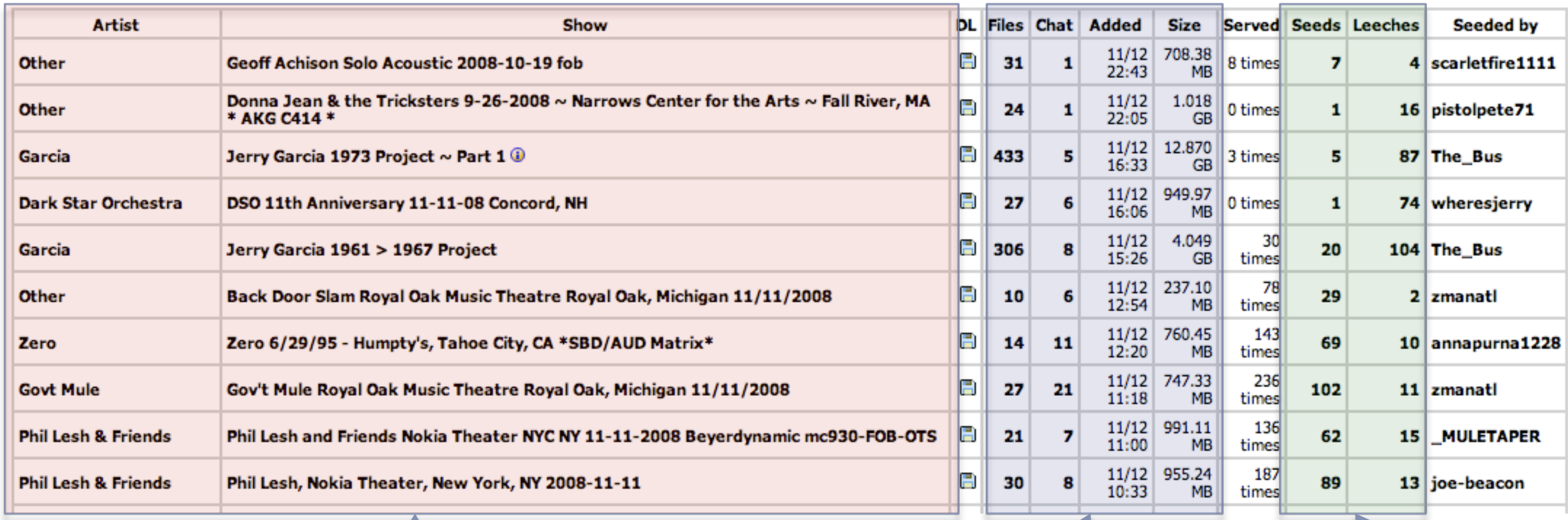
- One problem we have is that GUIDs are hashes, not human readable entities.
- Need a mapping of some human readable form to the GUID so a client can either find or provide the object.
- Systems like BitTorrent use "trackers" and web pages to help with this.



Artist	Show	DL	Files	Chat	Added	Size	Served	Seeds	Leeches	Seeded by
Other	Geoff Achison Solo Acoustic 2008-10-19 fob	💾	31	1	11/12 22:43	708.38 MB	8 times	7	4	scarletfire1111
Other	Donna Jean & the Tricksters 9-26-2008 ~ Narrows Center for the Arts ~ Fall River, MA * AKG C414 *	💾	24	1	11/12 22:05	1.018 GB	0 times	1	16	pistolpete71
Garcia	Jerry Garcia 1973 Project ~ Part 1 ⓘ	💾	433	5	11/12 16:33	12.870 GB	3 times	5	87	The_Bus
Dark Star Orchestra	DSO 11th Anniversary 11-11-08 Concord, NH	💾	27	6	11/12 16:06	949.97 MB	0 times	1	74	wheresjerry
Garcia	Jerry Garcia 1961 > 1967 Project	💾	306	8	11/12 15:26	4.049 GB	30 times	20	104	The_Bus
Other	Back Door Slam Royal Oak Music Theatre Royal Oak, Michigan 11/11/2008	💾	10	6	11/12 12:54	237.10 MB	78 times	29	2	zmanatl
Zero	Zero 6/29/95 - Humpty's, Tahoe City, CA *SBD/AUD Matrix*	💾	14	11	11/12 12:20	760.45 MB	143 times	69	10	annapurna1228
Govt Mule	Gov't Mule Royal Oak Music Theatre Royal Oak, Michigan 11/11/2008	💾	27	21	11/12 11:18	747.33 MB	236 times	102	11	zmanatl
Phil Lesh & Friends	Phil Lesh and Friends Nokia Theater NYC NY 11-11-2008 Beyerdynamic mc930-FOB-OTS	💾	21	7	11/12 11:00	991.11 MB	136 times	62	15	_MULETAPER
Phil Lesh & Friends	Phil Lesh, Nokia Theater, New York, NY 2008-11-11	💾	30	8	11/12 10:33	955.24 MB	187 times	89	13	joe-beacon

Trackers

- BT Trackers are interesting. Look at what information is here:



Artist	Show	DL	Files	Chat	Added	Size	Served	Seeds	Leeches	Seeded by
Other	Geoff Achison Solo Acoustic 2008-10-19 fob	💾	31	1	11/12 22:43	708.38 MB	8 times	7	4	scarletfire1111
Other	Donna Jean & the Tricksters 9-26-2008 ~ Narrows Center for the Arts ~ Fall River, MA * AKG C414 *	💾	24	1	11/12 22:05	1.018 GB	0 times	1	16	pistolpete71
Garcia	Jerry Garcia 1973 Project ~ Part 1 ⓘ	💾	433	5	11/12 16:33	12.870 GB	3 times	5	87	The_Bus
Dark Star Orchestra	DSO 11th Anniversary 11-11-08 Concord, NH	💾	27	6	11/12 16:06	949.97 MB	0 times	1	74	wheresjerry
Garcia	Jerry Garcia 1961 > 1967 Project	💾	306	8	11/12 15:26	4.049 GB	30 times	20	104	The_Bus
Other	Back Door Slam Royal Oak Music Theatre Royal Oak, Michigan 11/11/2008	💾	10	6	11/12 12:54	237.10 MB	78 times	29	2	zmanatl
Zero	Zero 6/29/95 - Humpty's, Tahoe City, CA *SBD/AUD Matrix*	💾	14	11	11/12 12:20	760.45 MB	143 times	69	10	annapurna1228
Govt Mule	Gov't Mule Royal Oak Music Theatre Royal Oak, Michigan 11/11/2008	💾	27	21	11/12 11:18	747.33 MB	236 times	102	11	zmanatl
Phil Lesh & Friends	Phil Lesh and Friends Nokia Theater NYC NY 11-11-2008 Beyerdynamic mc930-FOB-OTS	💾	21	7	11/12 11:00	991.11 MB	136 times	62	15	_MULETAPER
Phil Lesh & Friends	Phil Lesh, Nokia Theater, New York, NY 2008-11-11	💾	30	8	11/12 10:33	955.24 MB	187 times	89	13	joe-beacon

Name

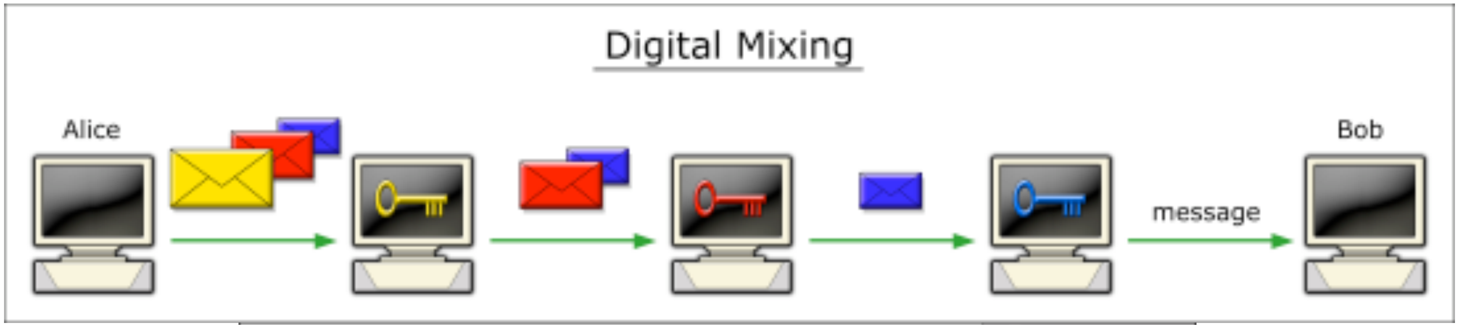
Size, content metadata

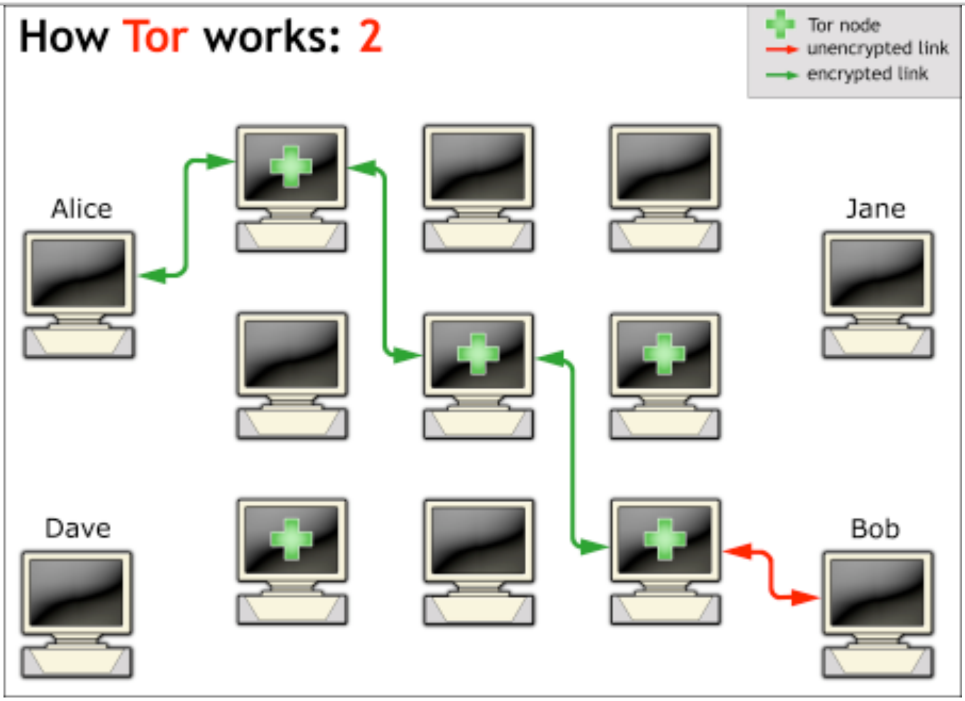
Seeds/Leechers: A reasonable metric of expected performance

Anonymity

- For various reasons, systems designers wish to provide anonymity within a system.
  - Typically achieved by routing methods within the P2P system.
- For example, the Vidalia/Tor project.
  - Proxy for anonymizing network activity.

Tor, Onion Routing (“Vidalia”)





***Source**: http://www.iusmentis.com/society/privacy/remailers/onionrouting/*

Case study: Pastry

- 128-bit GUID space

- Nodes have public keys, which are used to generate node GUIDs.
  - SHA-1 hashing.

- SHA-1 example:
  - “The quick brown fox jumps over the lazy dog”
    - **2fd4e1c6 7a2d28fc ed849ee1 bb76e739 1b93eb12**
  - "The quick brown fox jumps over the lazy **cog”**
    - **de9f2c7f d25e1b3a fad3e85a 0bd17d9b 100db4b3**

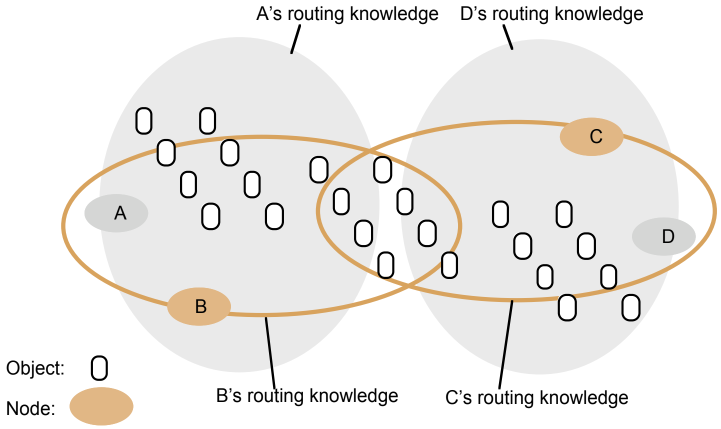
Pastry

- Given N nodes, a message to a GUID will take O(log N) steps.
- If GUID active, message will go to it.
- Otherwise, it will go to a message with a nearby GUID.

- Routing is a repeated process of getting closer and closer to the target GUID.
  - Metric is defined on GUIDs, not actual locations or IP-related locations.

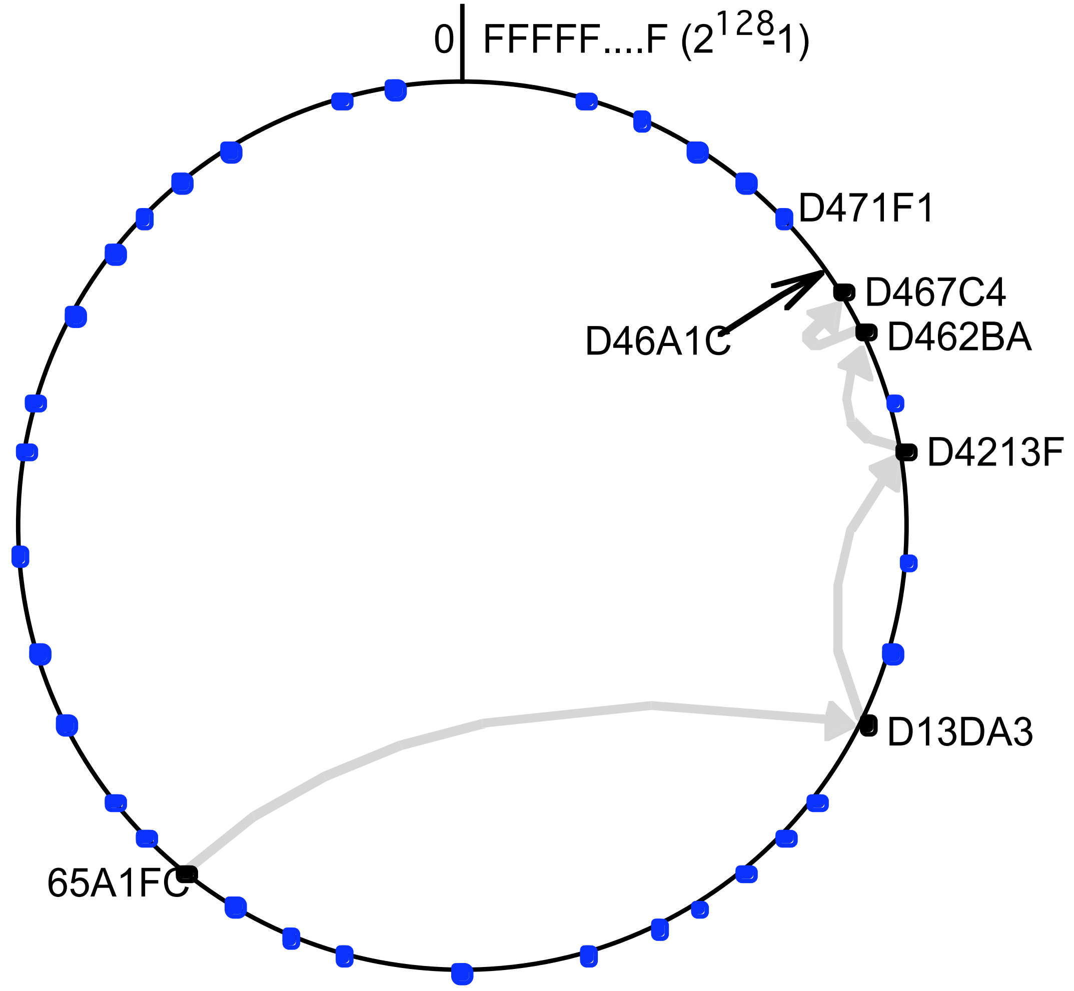
Pastry routing

- Nodes store a vector of GUIDs and IP addresses of L peers with nearest GUIDs called a leaf set.
  - These are used to route messages closer to their destination.



- Pastry GUID space is treated as circular.
  - The neighbors of 0 are 1 and 2^128-1
- Uses prefix routing.
  - Clearest when we look at the next slide.

Routing pictorially



Host integration

- Node computes it's own GUID.
- Sends this in a "join" message to a nearby (network-wise) Pastry node.
- This node then sends the join message onward towards the GUID of a node closest to that which is joining (GUID-wise).
- As the join message traverses the network, participants through which it is routed will send part of their routing information to the new node.
- The final node that receives the join message also provides it's leaf set as a seed for the new node.

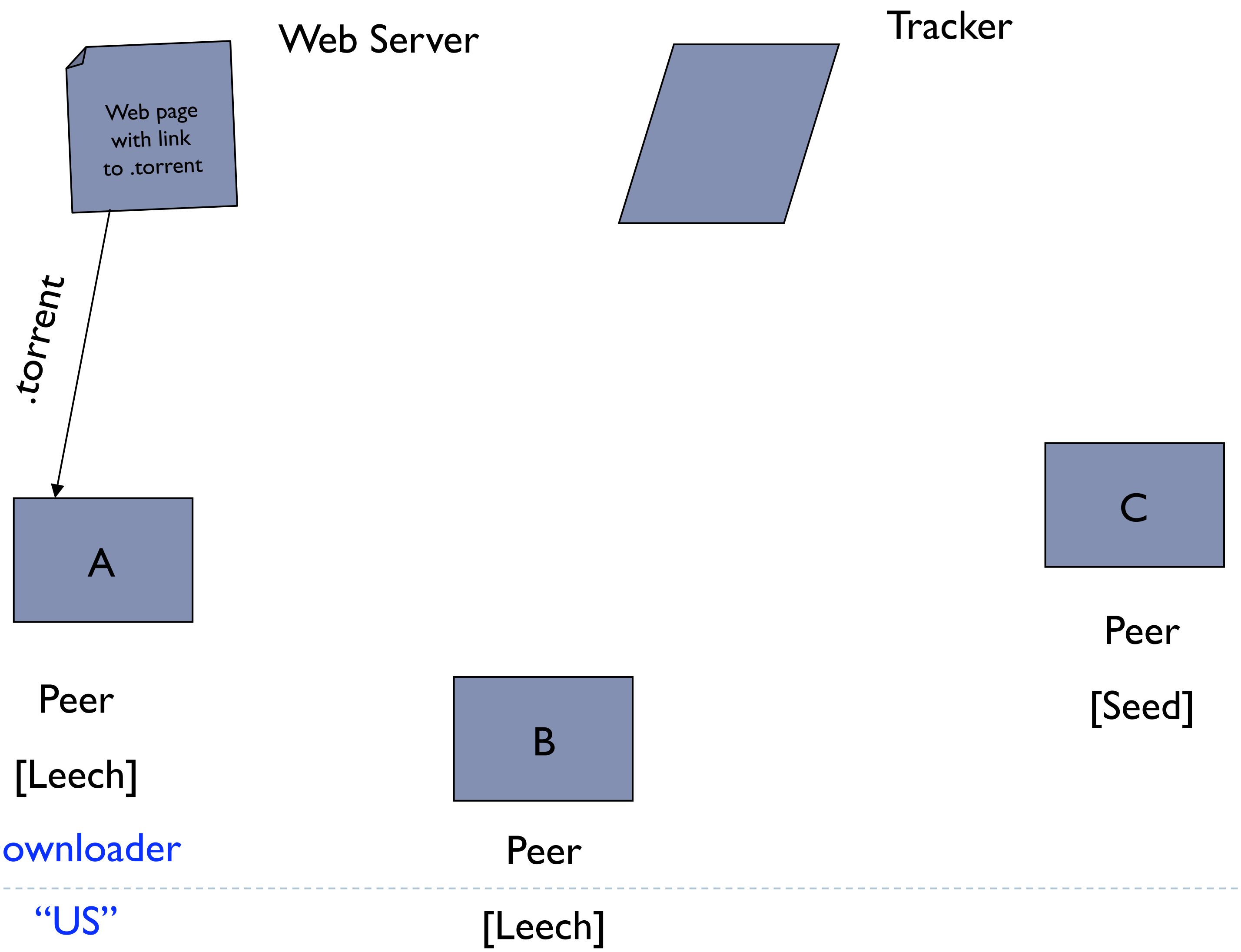
Host failure and fault tolerance

- Say a node vanishes. Need to repair leaf sets of nodes that included it in theirs.
- Repair involves asking a node that is still alive that had a GUID near the failed node for it's leaf set.
  - High likelihood that there is overlap in the leaf set.
- Tolerates simple failures of single nodes or small node counts.
- Can't deal with failure counts simultaneously of approximately the same size as the leaf set.

BitTorrent

- File sharing.
- Based on every node holding a portion of a file.
- Whole file is slowly built up out of pieces.
- A participant can provide a piece as soon as it has downloaded it from another peer.

Overall Architecture



Web page with link to .torrent

Web Server

Tracker

.torrent

A

Peer

[Leech]

Downloader

“US”

B

Peer

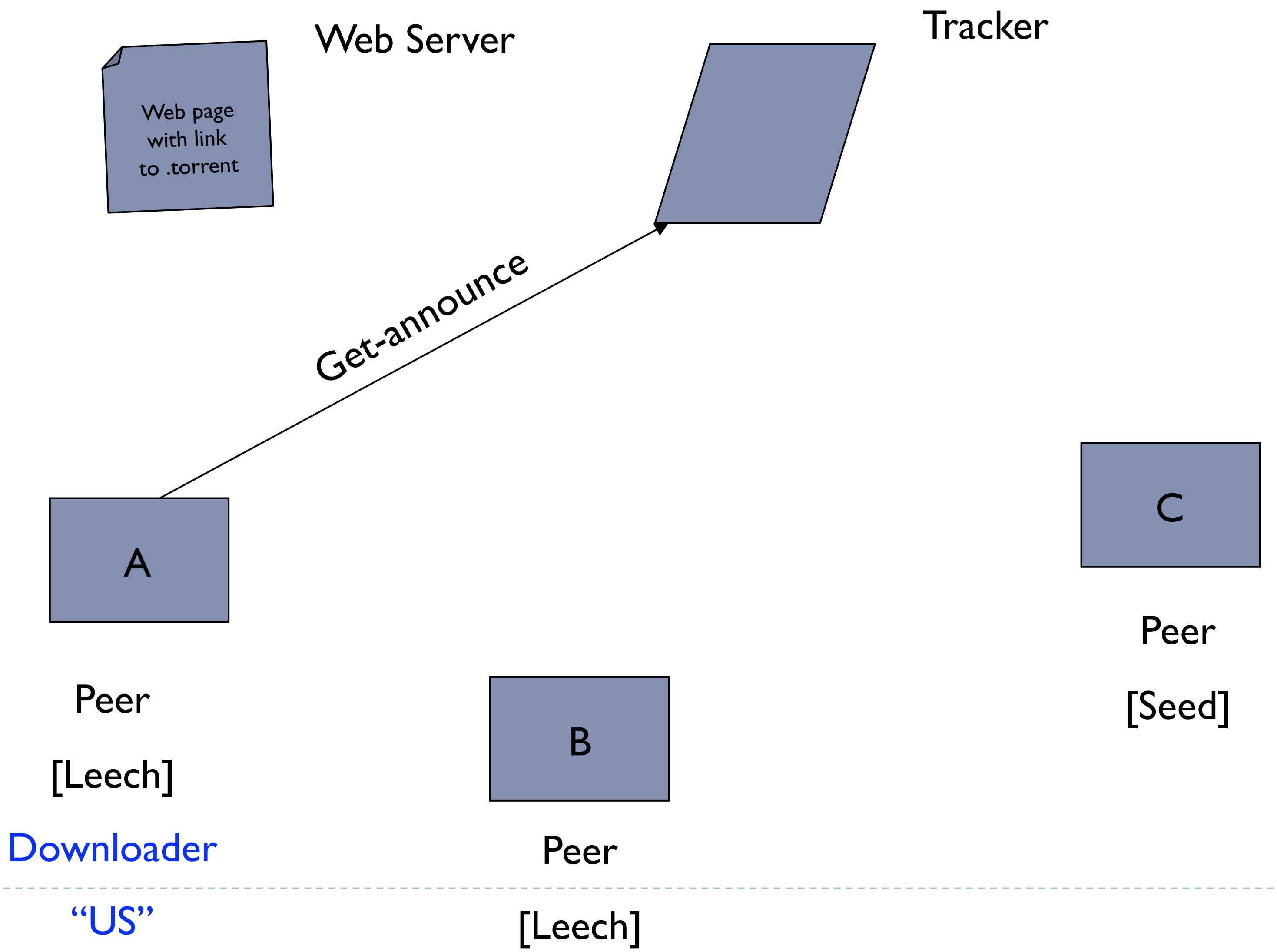
[Leech]

C

Peer

[Seed]

Overall Architecture



Web page with link to .torrent

Web Server

Tracker

Get-announce

A

Peer

[Leech]

Downloader

"US"

B

Peer

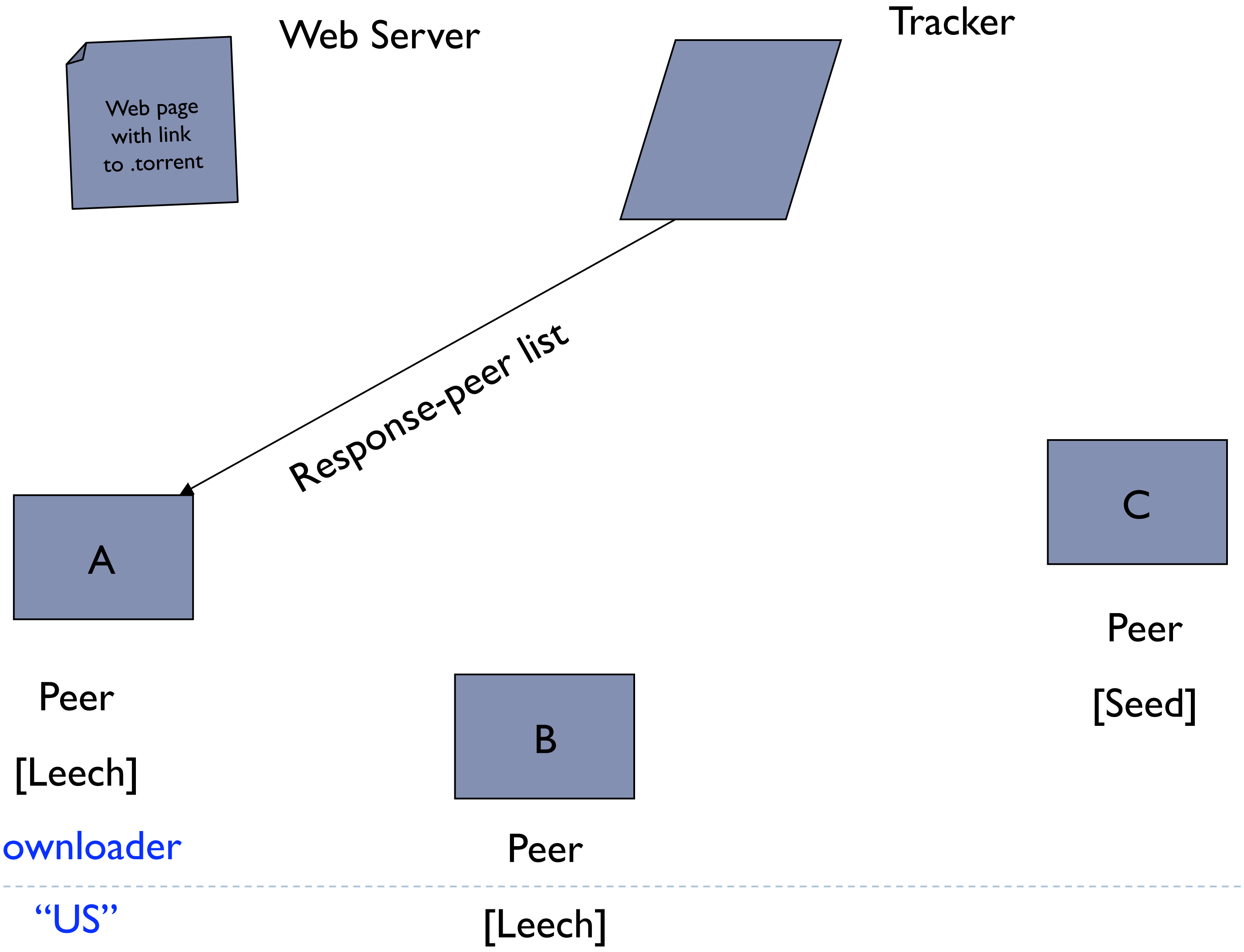
[Leech]

C

Peer

[Seed]

Overall Architecture



Web Server

Web page with link to .torrent

Tracker

Response-peer list

A

Peer

[Leech]

Downloader

"US"

B

Peer

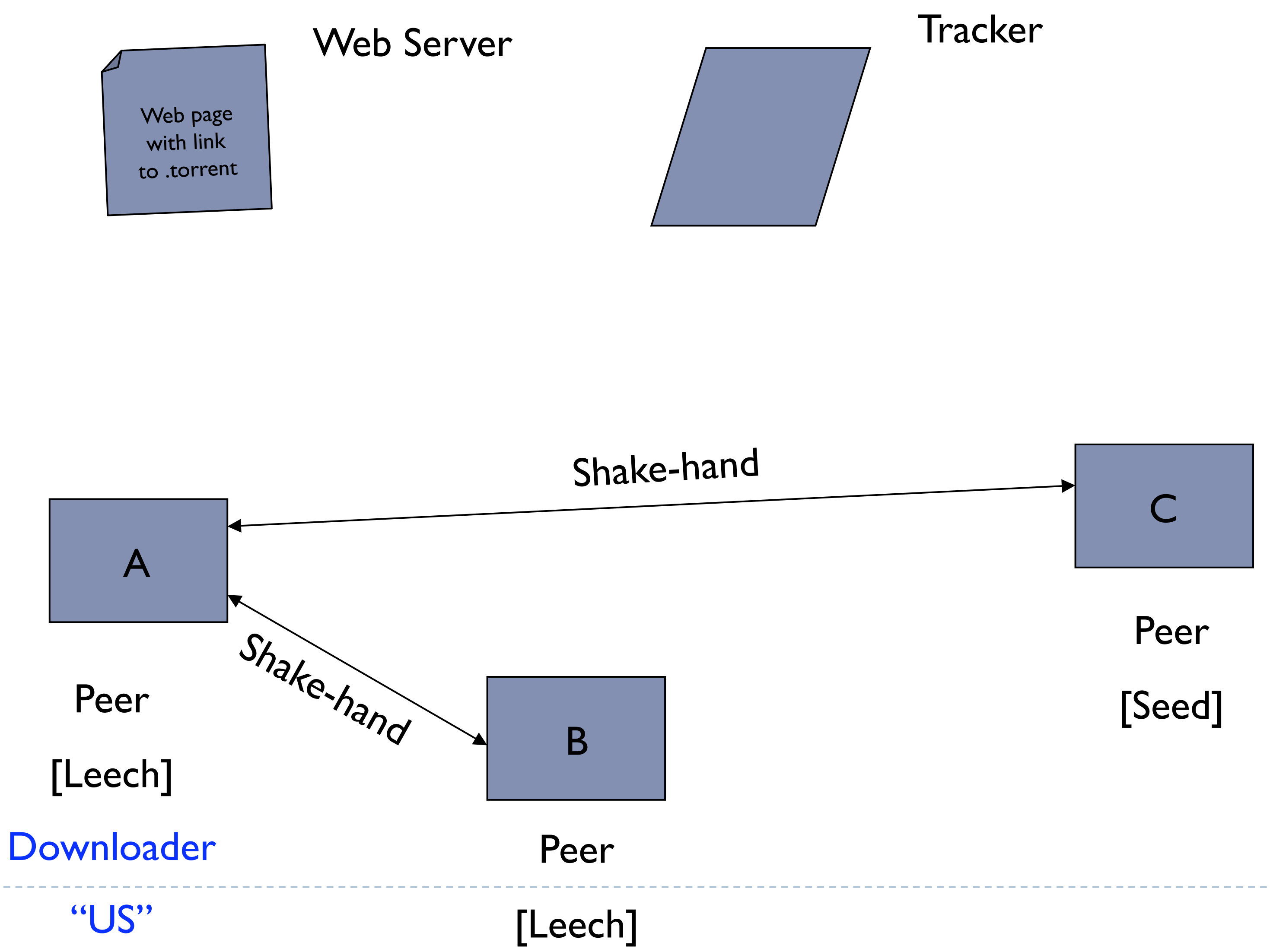
[Leech]

C

Peer

[Seed]

Overall Architecture



Web page with link to .torrent

Web Server

Tracker

Shake-hand

Shake-hand

A

B

C

Peer

[Leech]

Downloader

"US"

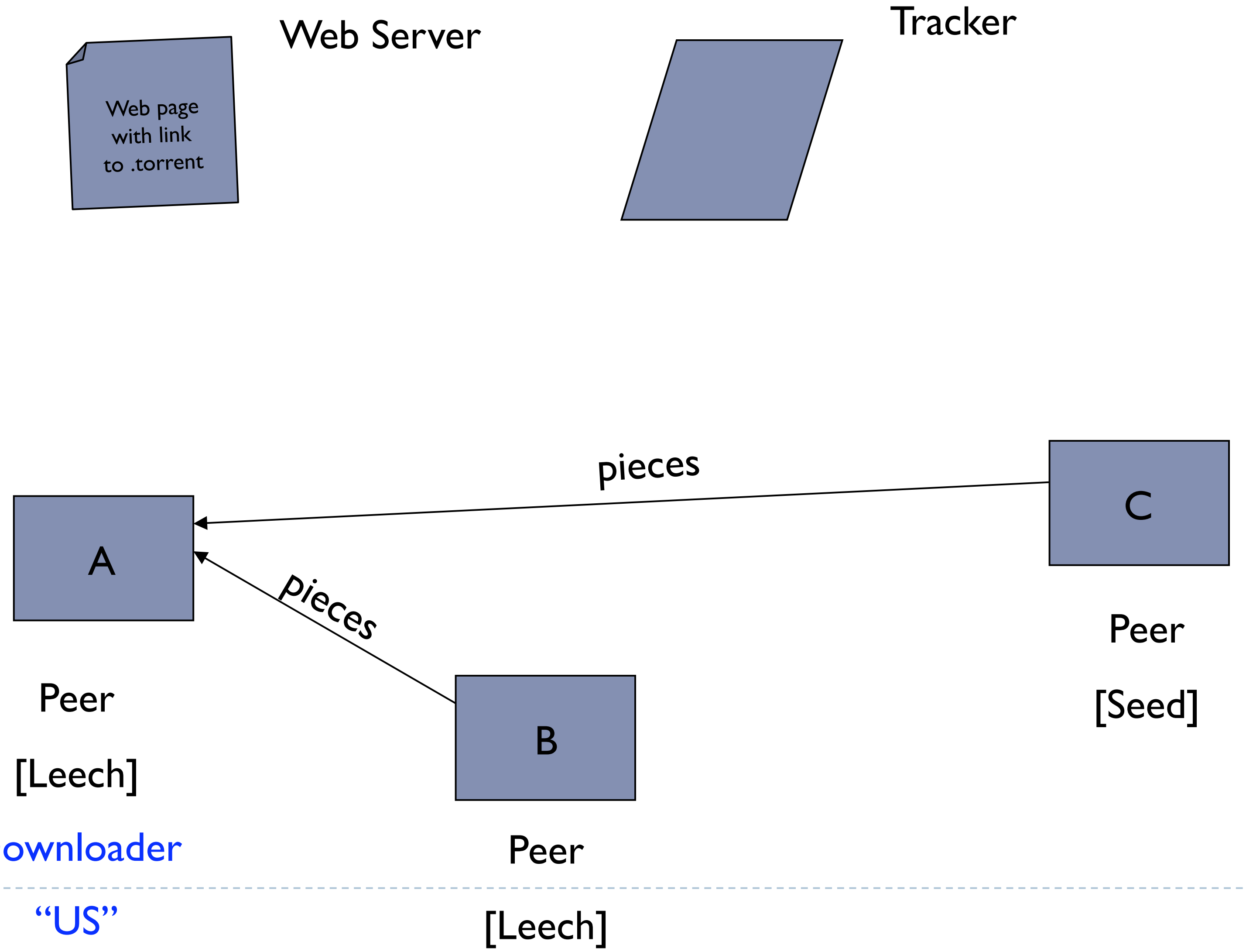
Peer

[Leech]

Peer

[Seed]

Overall Architecture



Web page with link to .torrent

Web Server

Tracker

pieces

pieces

A

Peer

[Leech]

Downloader

"US"

B

Peer

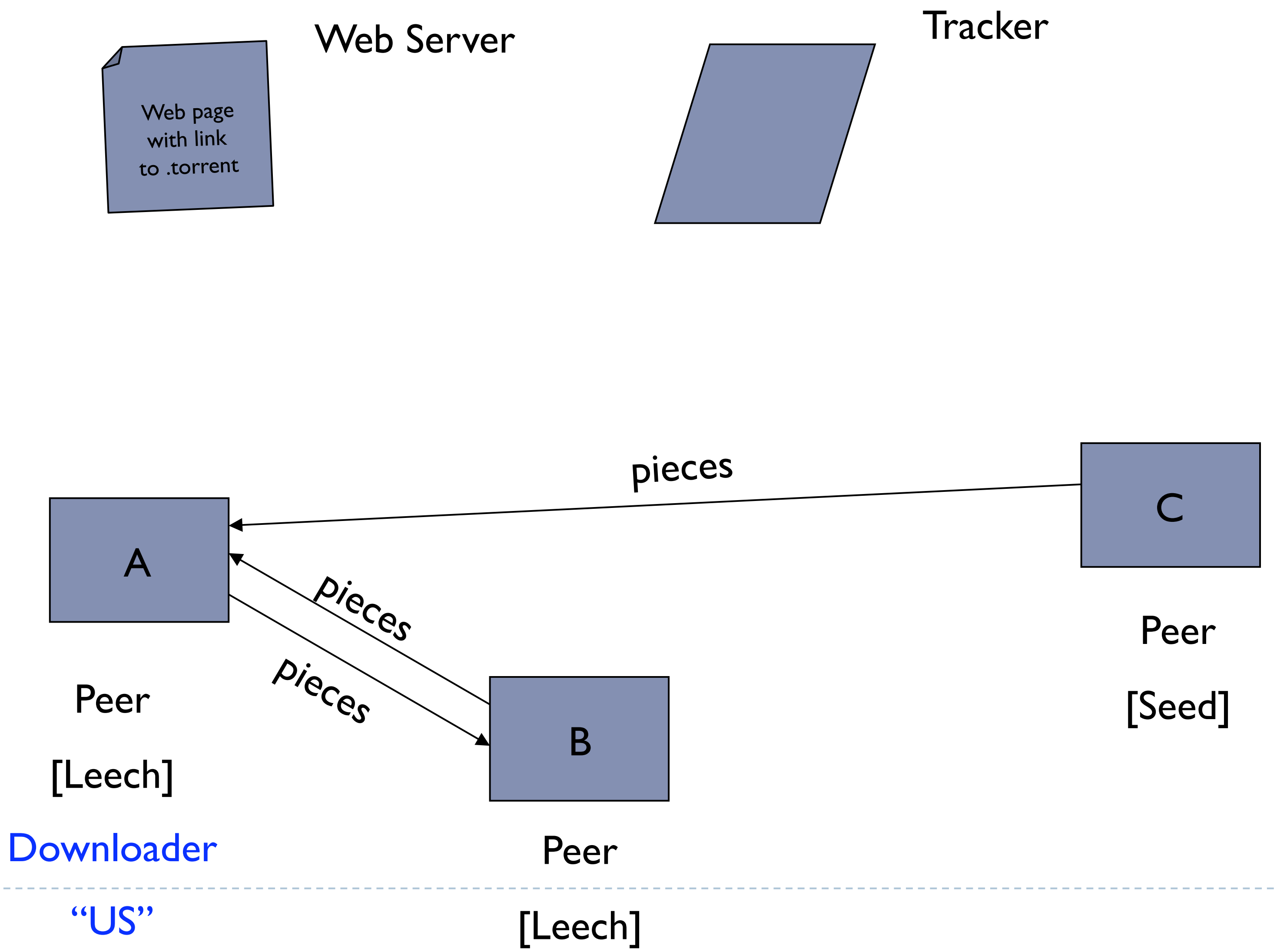
[Leech]

C

Peer

[Seed]

Overall Architecture



Web page with link to .torrent

Web Server

Tracker

pieces

C

A

pieces

pieces

B

Peer

[Seed]

Peer

[Leech]

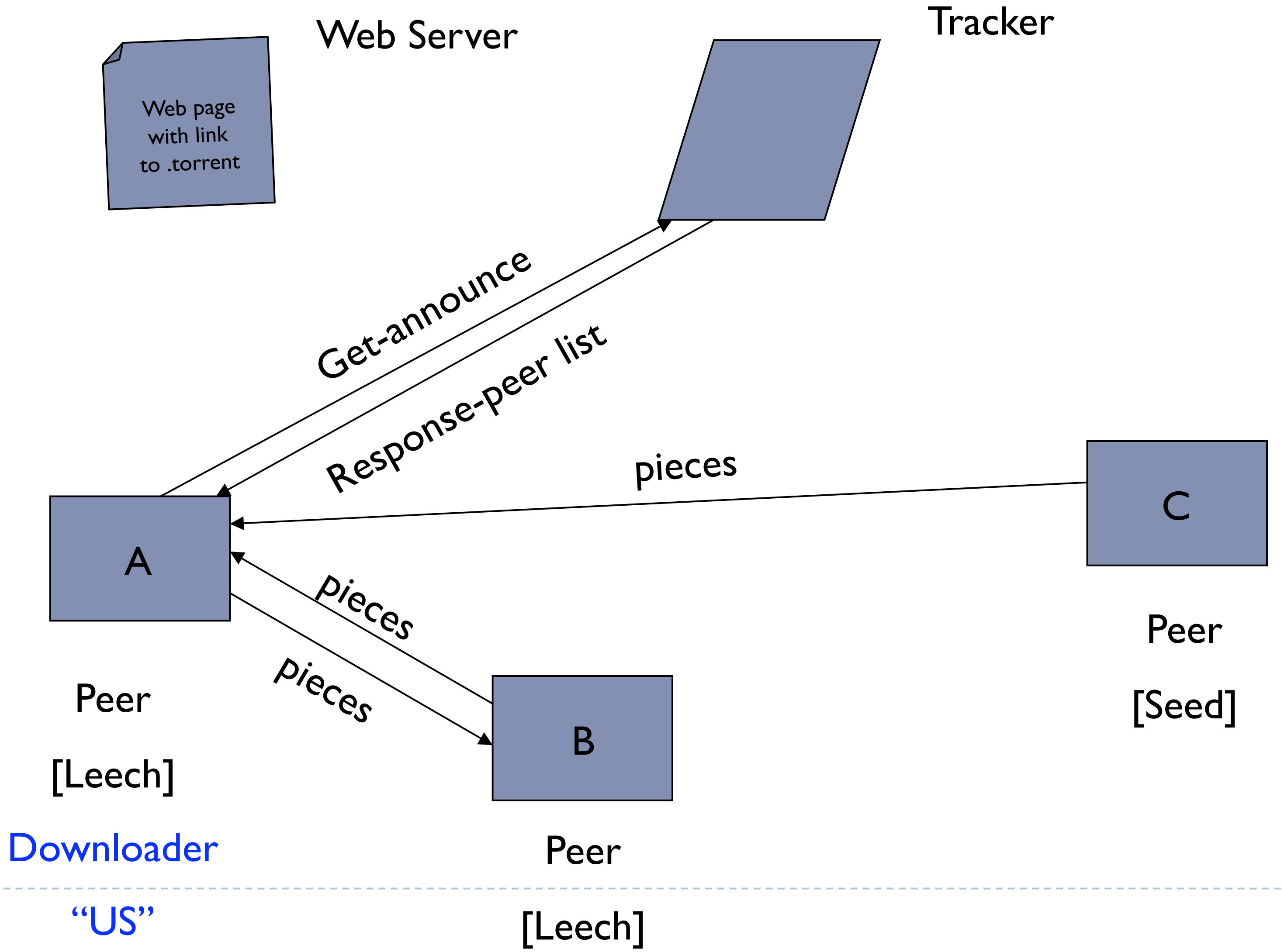
Downloader

"US"

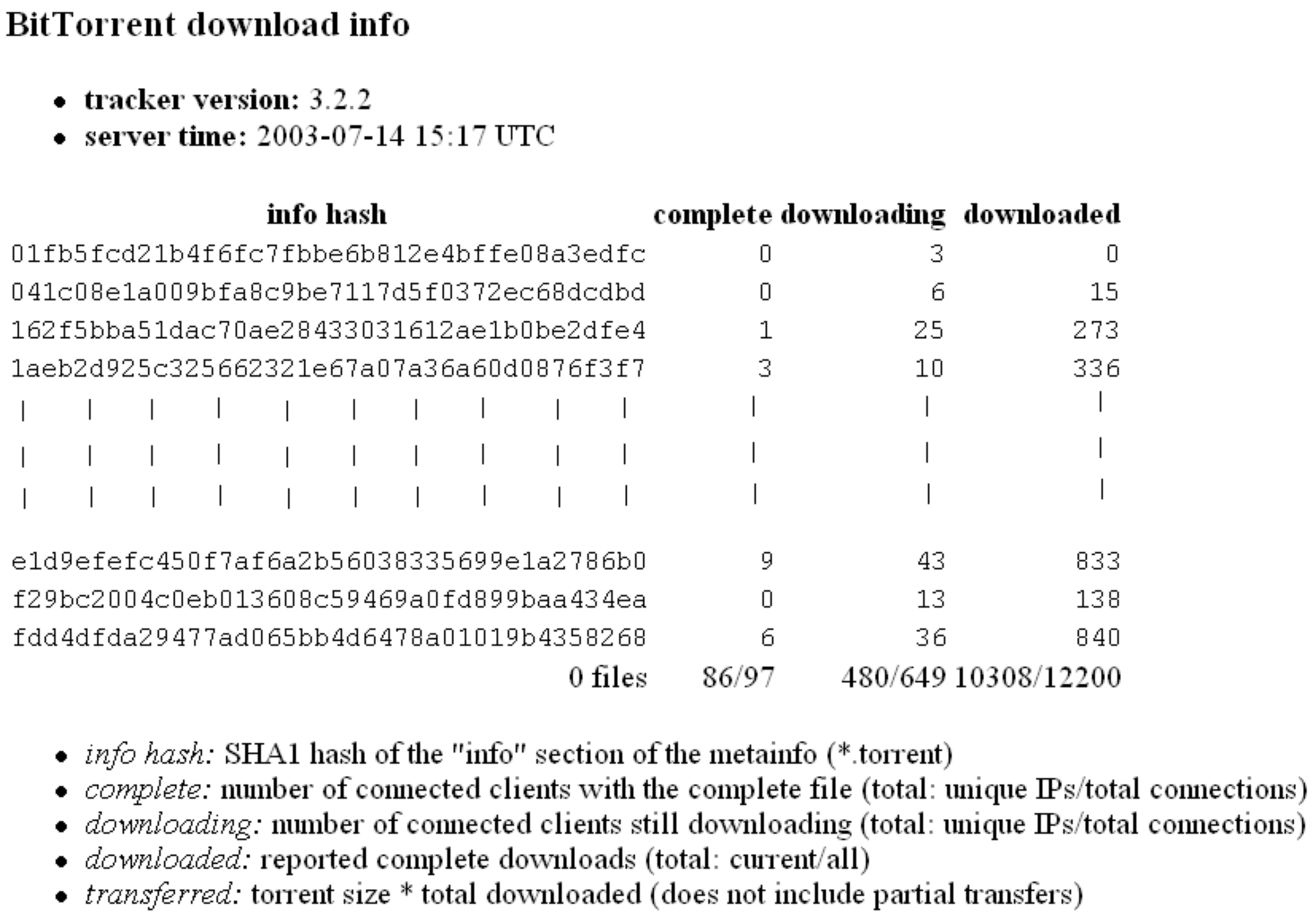
Peer

[Leech]

Overall Architecture



Tracker information



BitTorrent download info

- **tracker version:** 3.2.2
- **server time:** 2003-07-14 15:17 UTC

info hash	complete	downloading	downloaded
01fb5fcd21b4f6fc7fbbe6b812e4bffe08a3edfc	0	3	0
041c08e1a009bfa8c9be7117d5f0372ec68dcdbd	0	6	15
162f5bba51dac70ae28433031612ae1b0be2dfe4	1	25	273
1aeb2d925c325662321e67a07a36a60d0876f3f7	3	10	336
│ │ │ │ │ │ │ │ │ │	│	│	│
│ │ │ │ │ │ │ │ │ │	│	│	│
│ │ │ │ │ │ │ │ │ │	│	│	│
e1d9efefc450f7af6a2b56038335699e1a2786b0	9	43	833
f29bc2004c0eb013608c59469a0fd899baa434ea	0	13	138
fdd4dfda29477ad065bb4d6478a01019b4358268	6	36	840
0 files	86/97	480/649	10308/12200

- *info hash:* SHA1 hash of the "info" section of the metainfo (*.torrent)
- *complete:* number of connected clients with the complete file (total: unique IPs/total connections)
- *downloading:* number of connected clients still downloading (total: unique IPs/total connections)
- *downloaded:* reported complete downloads (total: current/all)
- *transferred:* torrent size * total downloaded (does not include partial transfers)

BitTorrent uses

- Other than the obvious things-you-shouldn't-do, it has found some interesting real world applications.

- Not uncommon to get Linux distro's via BitTorrent faster than traditional single-server HTTP download for ISOs.

- World of Warcraft and other systems use BT to distribute updates and patches.
  - I assume this is because it is cheaper for the game company to defer the bandwidth costs to their users when updates are released.
  - Very useful when an app has a massive user community, like WoW.

Peer to peer wrapup

- Active area of research.
- The big areas of research are related to:
  - Routing algorithms
  - Anonymity: both providing, and removing.
- Goals are load balance and idle resource utilization.

- Unfortunately, peer-to-peer has gained an unfair reputation due to the anonymity and sharing that it provides.
  - It's first popular uses were illegal, and that continues with current generation systems.
  - Important note: See above. One active area of research is detection – identifying participants in P2P networks, and identifying what they share.
    - Legal implications here – beware the temptation of PirateBay and other sites.