BitTorrent

Slides adapted from

Prof. Dah Ming Chiu, Chinese University of Hong Kong, Hong Kong

Dr. Iqbal Mohomed, University of Toronto, Canada

Content Distribution

- IP multicast
- CDN (Content Distribution Network)
- Application layer multicast
  - Overlay structures
    - Tree-based (push)
    - Data-driven (pull)
  - P2P swarming
    - BitTorrent, CoolStreaming

BitTorrent

- Released in the summer of 2001
- Basic ideas from game theory to largely eliminate the free-rider problem
  - All precedent systems could not deal with this problem well
- No strong guarantees unlike DHTs
- Working extremely well in practice unlike DHTs ☺

Basic Idea – Swarming Protocol

- A file is chopped into small pieces, called chunks
- Pieces are disseminated over the network
- As soon as a peer acquire a piece, it can trade it for missing pieces with other peers
- A peer hopes to be able to assemble the entire file at the end

Basic Components

- Web server
- The .torrent file
- Tracker
- Peers

Web Server

- Content discovery (i.e., file search) is handled outside of BitTorrent, using a Web server
  - To provide the “meta-info” file by HTTP
  - For example, http://bt.btchina.net
- The information about each movie or content is stored in a metafile such as “supergirl.torrent”

The .torrent File

- Static file storing necessary meta information
  - Name
  - Size
  - Checksum
    - The content is divided into many “chunks” (e.g., 1/4 megabyte each)
    - Each chunk is hashed to a checksum value
    - When a peer later gets the chunks (from other peers), it can check the authenticity by comparing the checksum
  - IP address and port of the **Tracker**

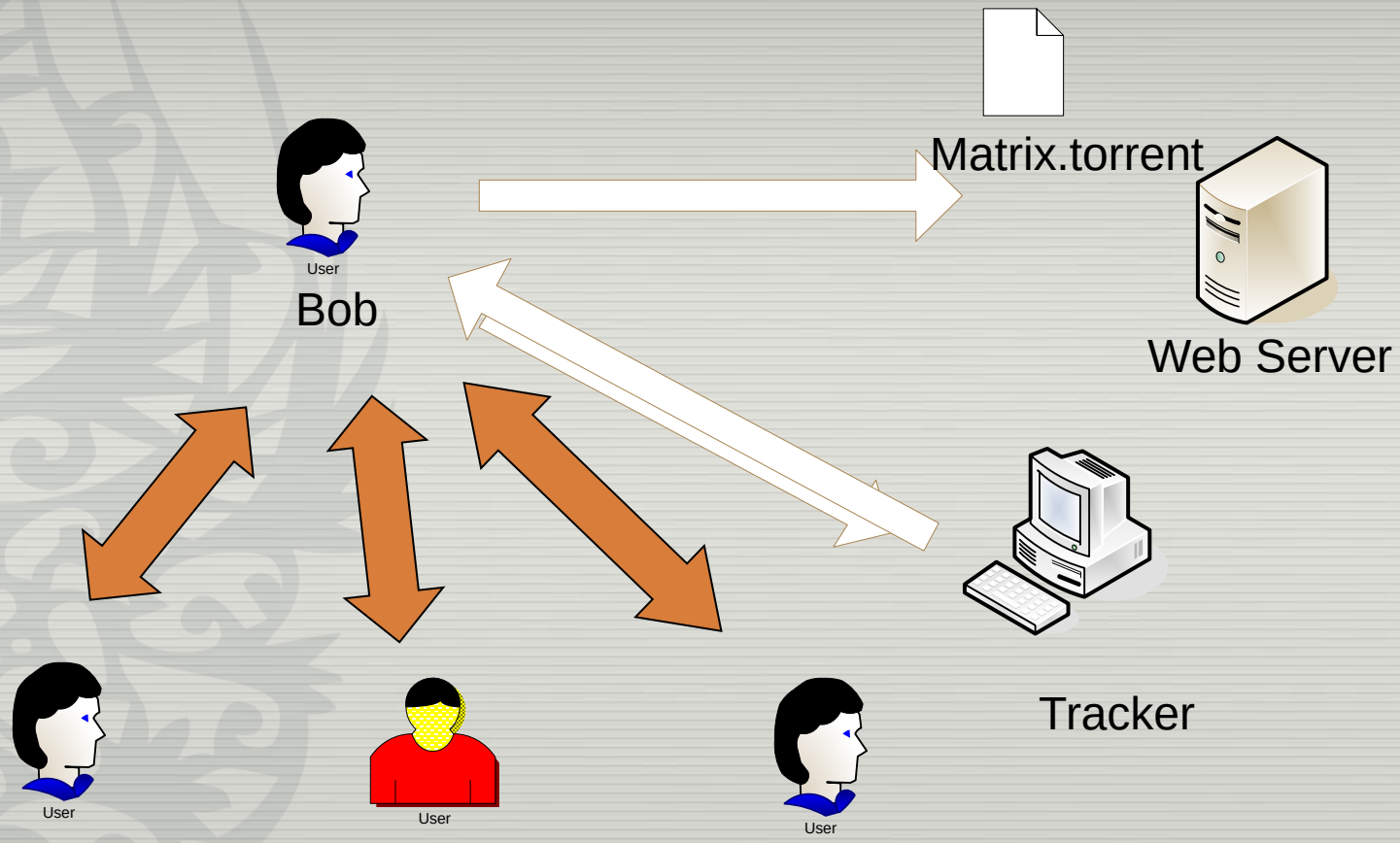
Tracker

- Keeping track of peers
  - To allow peers to find one another
  - To return a random list of active peers

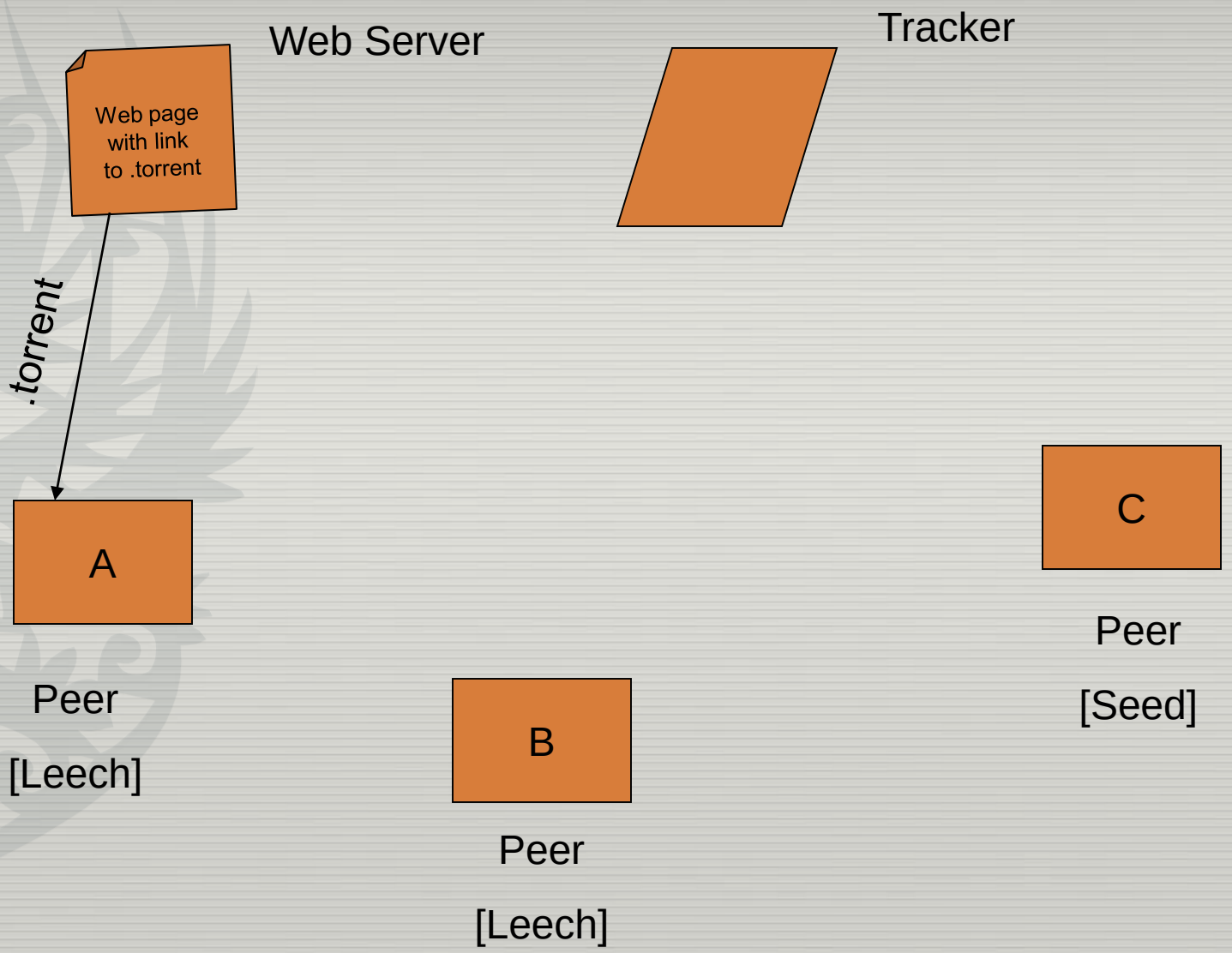
Peers

- Two types of peers:
  - *Downloader* (*leecher*) : A peer who has only ***a part (or none)*** of the file.
  - *Seeder*: A peer who has the ***complete*** file, and chooses to stay in the system to allow other peers to download

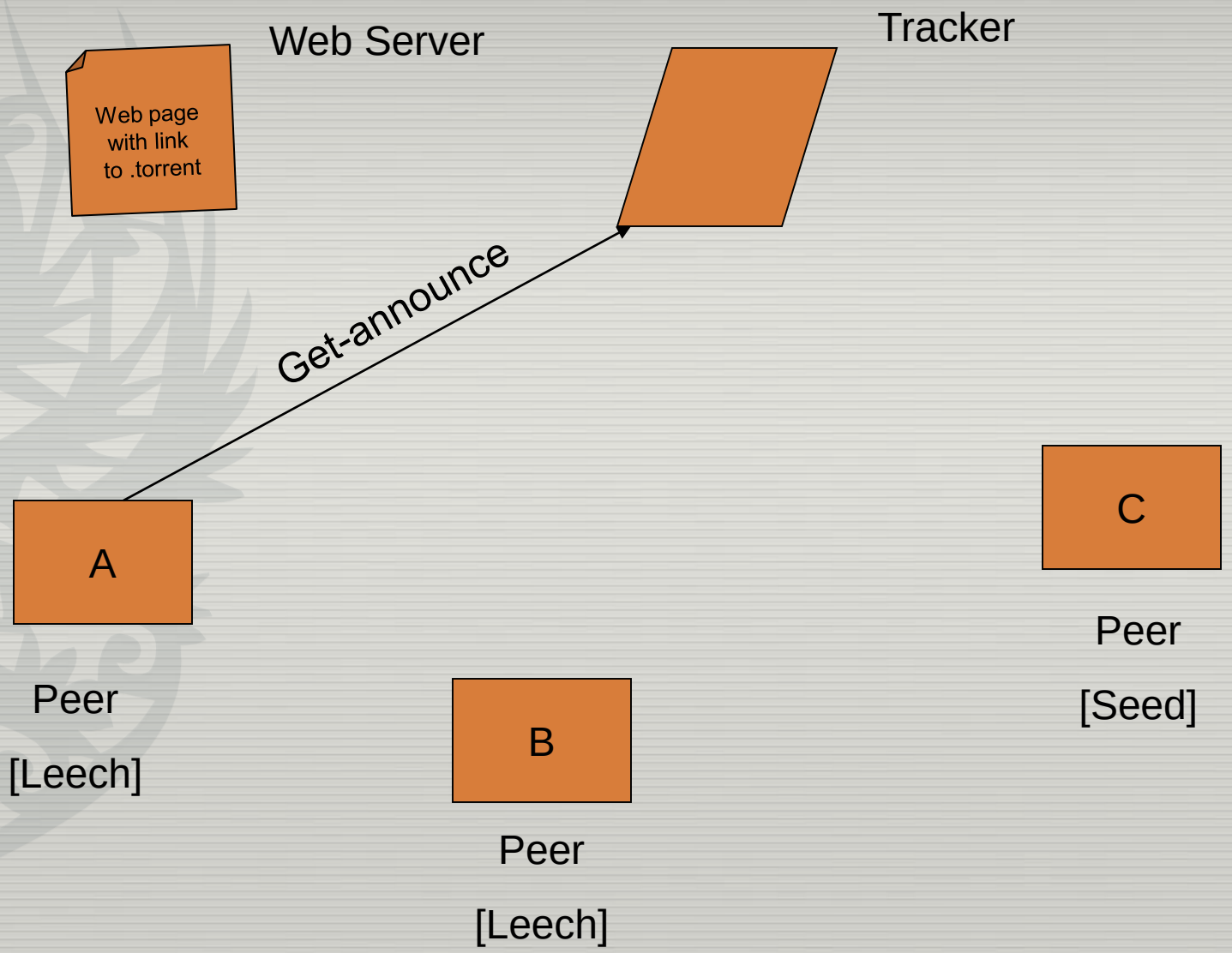
BitTorrent in Action



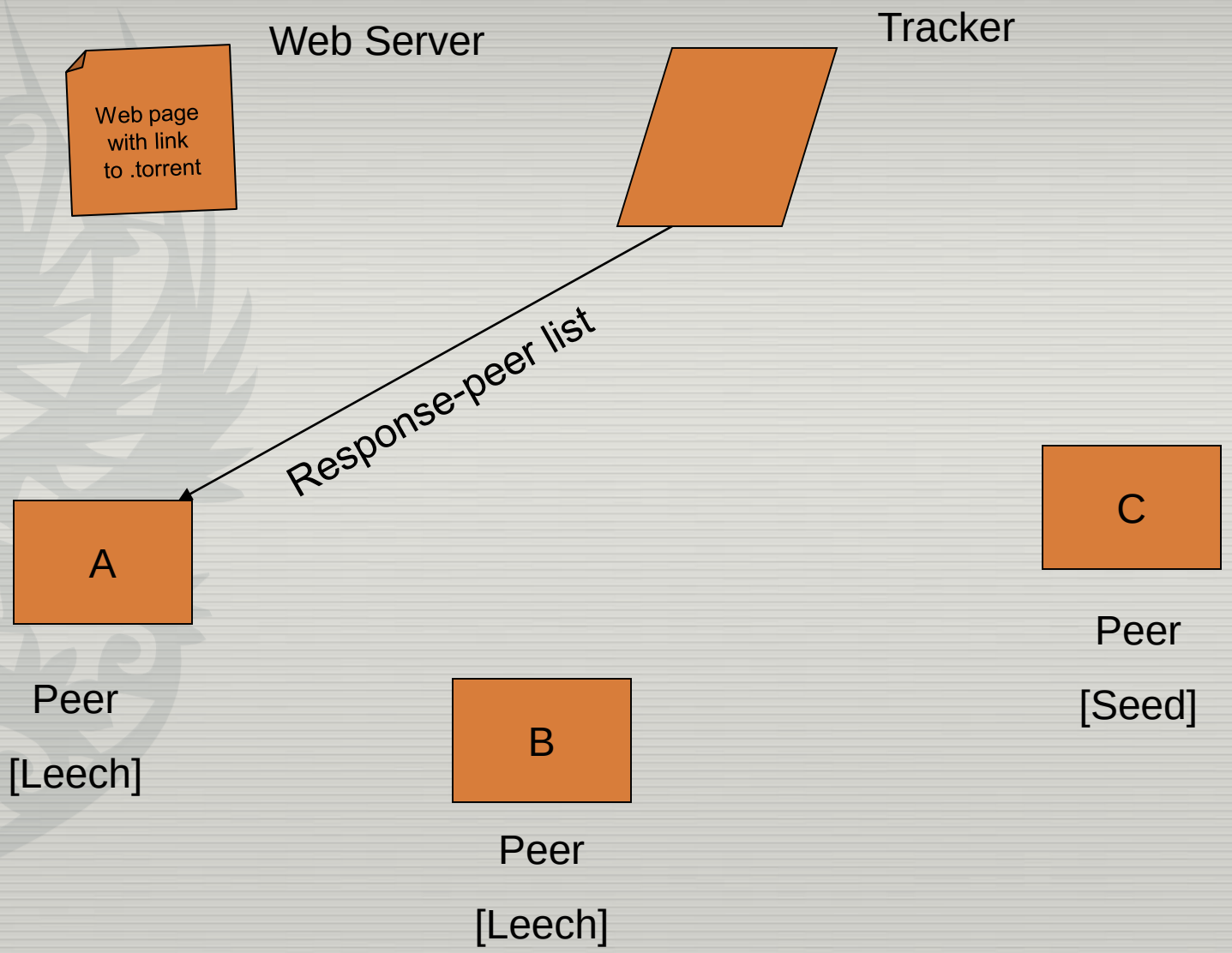
Overview – System Components



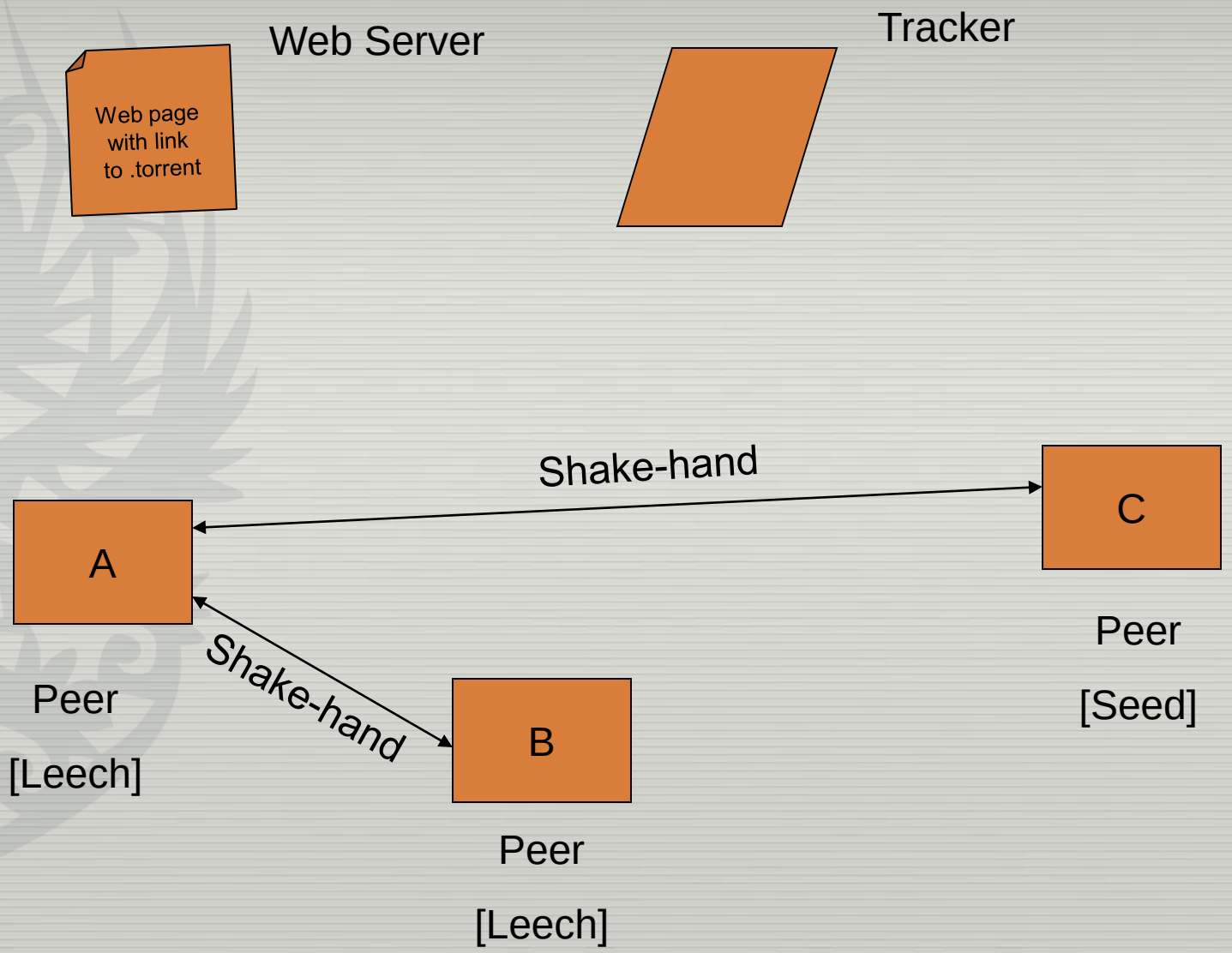
Overview – System Components



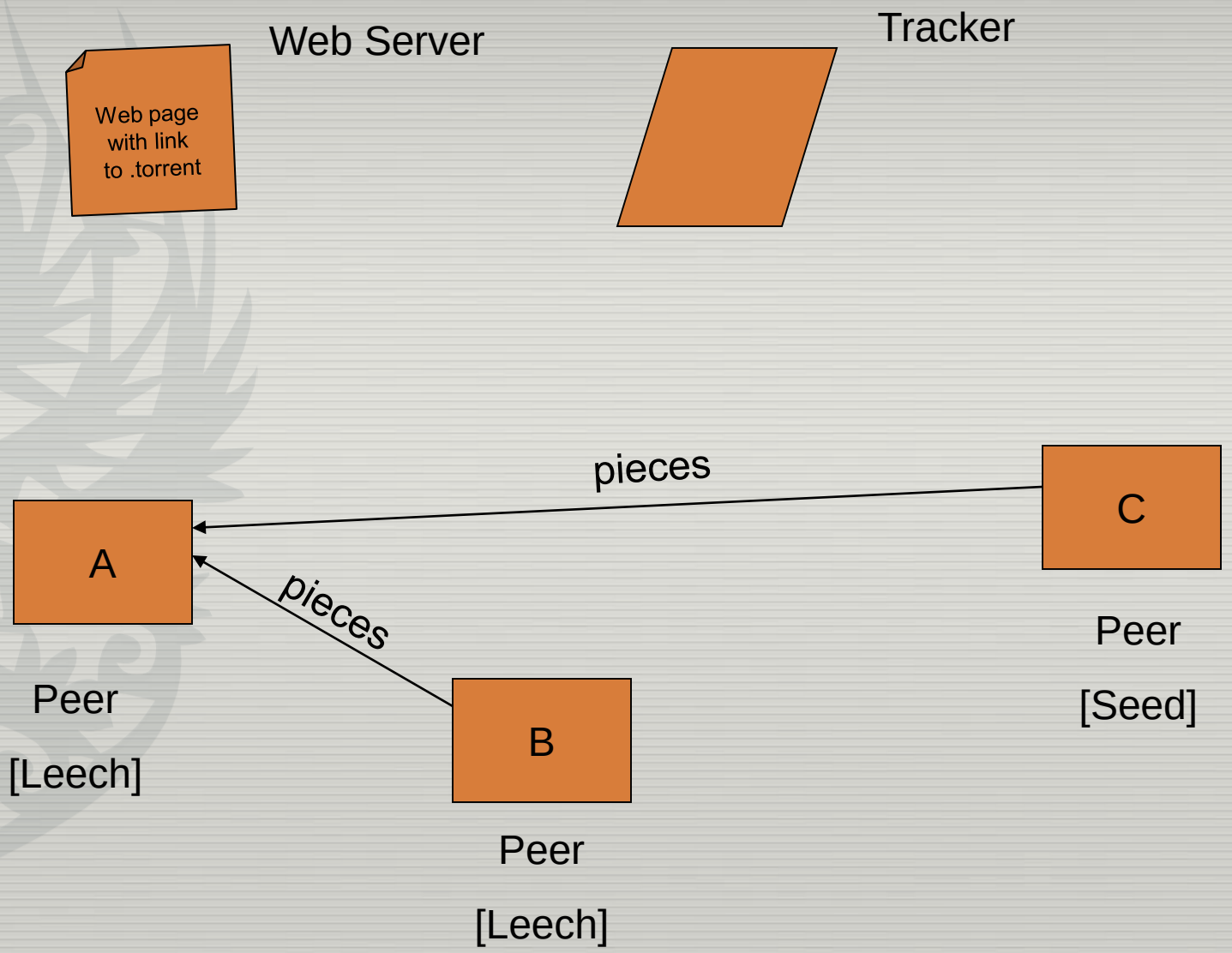
Overview – System Components



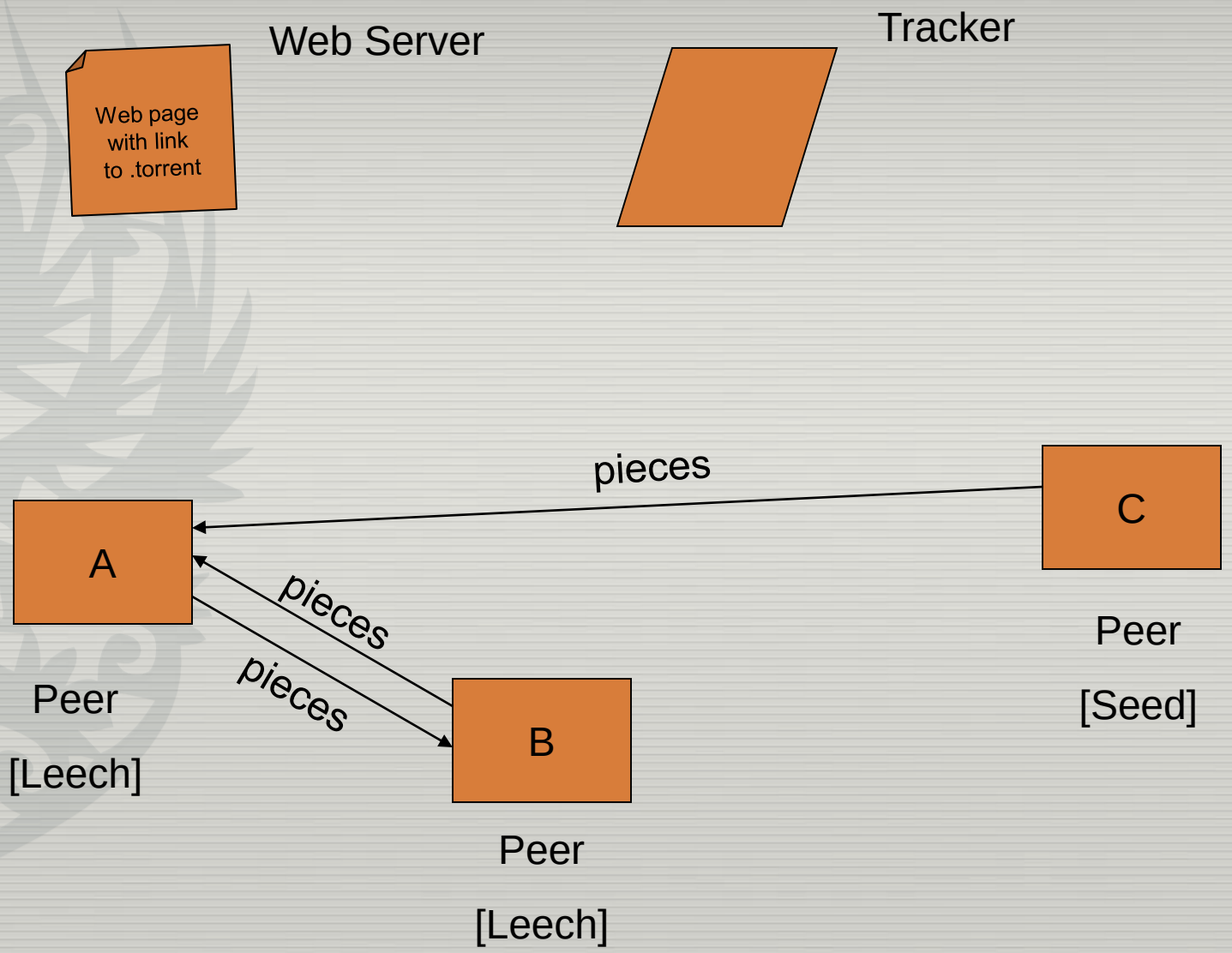
Overview – System Components



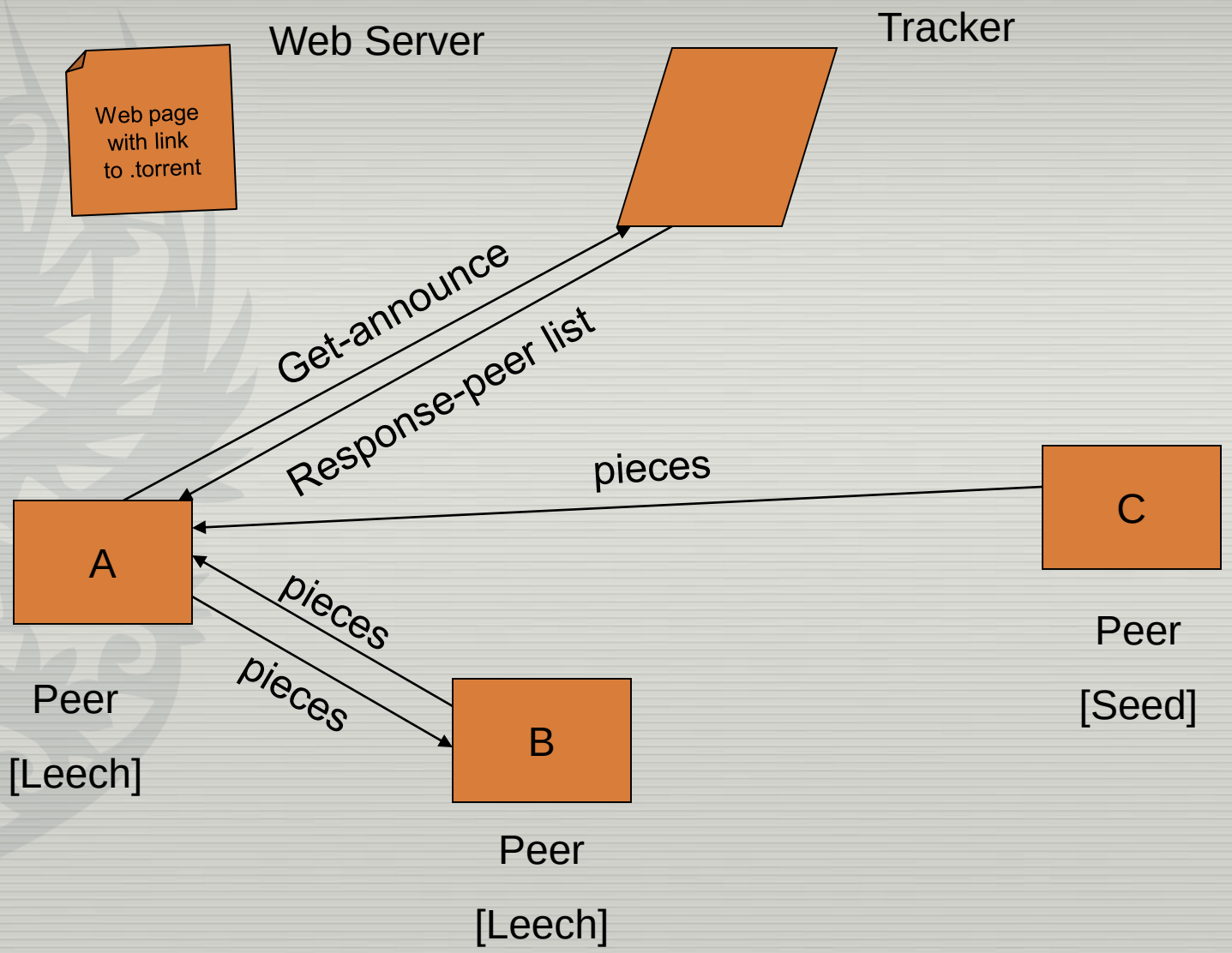
Overview – System Components



Overview – System Components



Overview – System Components

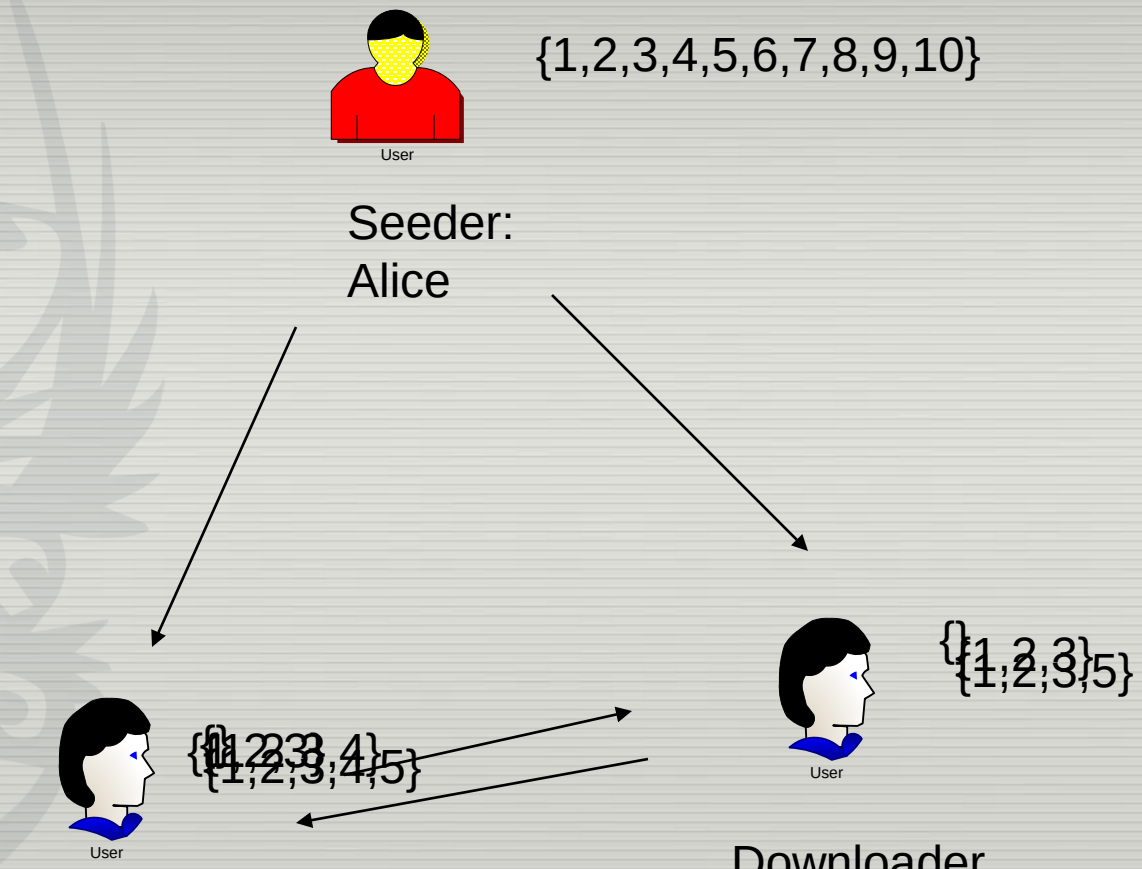


Chunks

- A file is split into chunks of fixed size, typically 256Kb
- Each peer maintains a bit map that indicates which chunks it has
- Each peer reports to all of its neighboring peers (obtained from tracker) what chunks it has
  - This is the information used to build the implicit delivery trees

피어는 주기적으로 bitmap을 교환한다. bitmap은 청크를 가지고 있는지 여부를 판단할수 있다.
bitmap은 파일 내에서의 특정한 청크를 나타낸다. bit==1이다면 그 청크를 가지고 있다
0이면 없음. 그 피어가 어떤 청크를 가지고 있는지 확인해서 내가 없는 청크를 가진 peer에게
요청을 보낸다.

Swarming Example



Rarest First

어떤 전략으로 chunk요청?
missing chunk가 여러개니까
자기 neighbor들이 가진 chunk들을 보고
가장 희귀한 chunk부터 요청한다.

- Rarer pieces are given priority in downloading with the rarest being the first candidate
- The most common pieces are postponed towards the end
- This policy ensures that a variety of pieces are downloaded from the seeder, resulting in quicker chunk propagation

quicker chunk propagation = 희귀한 청크를 공급하는
피어가 하나 늘어나니까 전체적 네트워크에서보면
파일의 확산에 도움이 된다.
(청크를 제공하는 사람이 많을수록
속도가 빨라지니까)
+ 희귀한 청크를 가진 사용자가 나가버리면 availability가
사라지니까.

Peer Selection

peer를 어떻게 선택하는가?
고정된것이 아니라 계속 업데이트된다.
chunk를 주고받을 peer를 어떻게 선택할 것인가?

Basic idea of **tit-for-tat** strategy in BitTorrent:

- Maintain 4-5 "friends" with which to exchange chunks
- If a friend is not exchanging enough chunks, get rid of him/her
  - Known as "choking" in BT
- Periodically, randomly select a new friend
  - Known as "optimistic unchoking" in BT
- If you have no friends, randomly select several new friends
  - Known as "anti-snubbing" in BT

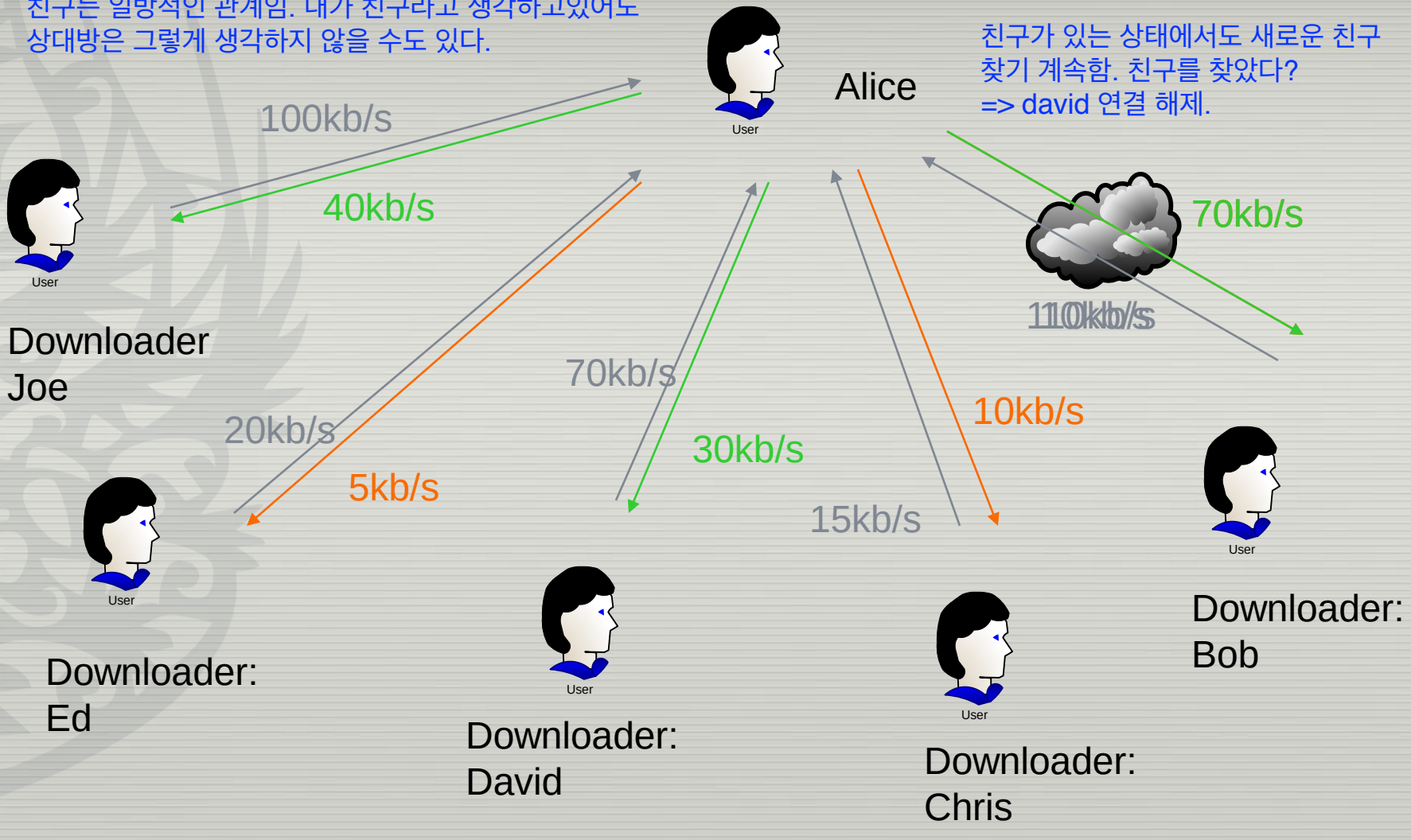
tit-for-tat = 받은만큼 돌려준다.
chunk를 많이 보내준다 = 친구
chunk를 서로 보내주지 않음 = 친구해제
계속 시험을 함.

일단 좋은 관계를 가정하고
좋은 속도로 chunk를 줘본다 => 잘주면
친구목록 추가 아니면 다른애 찾기

Example of Optimistic Unchoking

tit-for-tat = 내가 잘해줄수록
다른 사람이 잘해주는 인센티브제도.
친구는 일방적인 관계임. 내가 친구라고 생각하고있어도
상대방은 그렇게 생각하지 않을 수도 있다.

친구가 있는 상태에서도 새로운 친구
찾기 계속함. 친구를 찾았다?
=> david 연결 해제.



Questions?