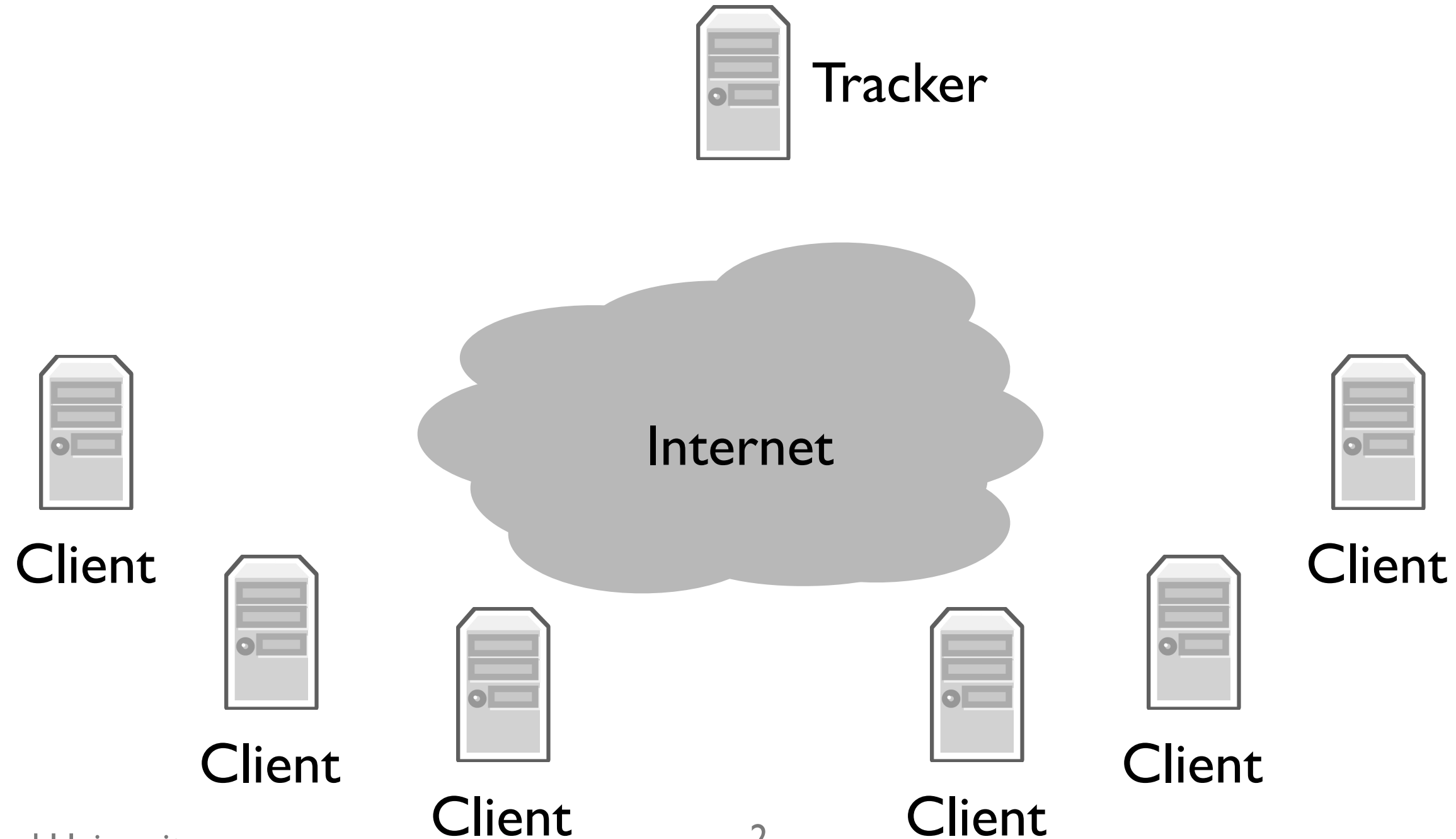


# BitTorrent

Swarms, Rarest-First and Tit-for-Tat

# BitTorrent



# Torrent File

- Torrent file (.torrent) describes file to download
  - ▶ Names tracker, server tracking who is participating
  - ▶ File length, piece length, SHA1 hashes of pieces
  - ▶ Additional metadata (who created torrent, etc.)
  - ▶ Also specifies tracker
- Client contacts tracker, starts communicating with peers
- “Trackerless” torrents use something called a DHT (distributed hash table)
  - ▶ Information on swarm stored across many nodes
  - ▶ A distributed coordination mechanism

# Torrent Files

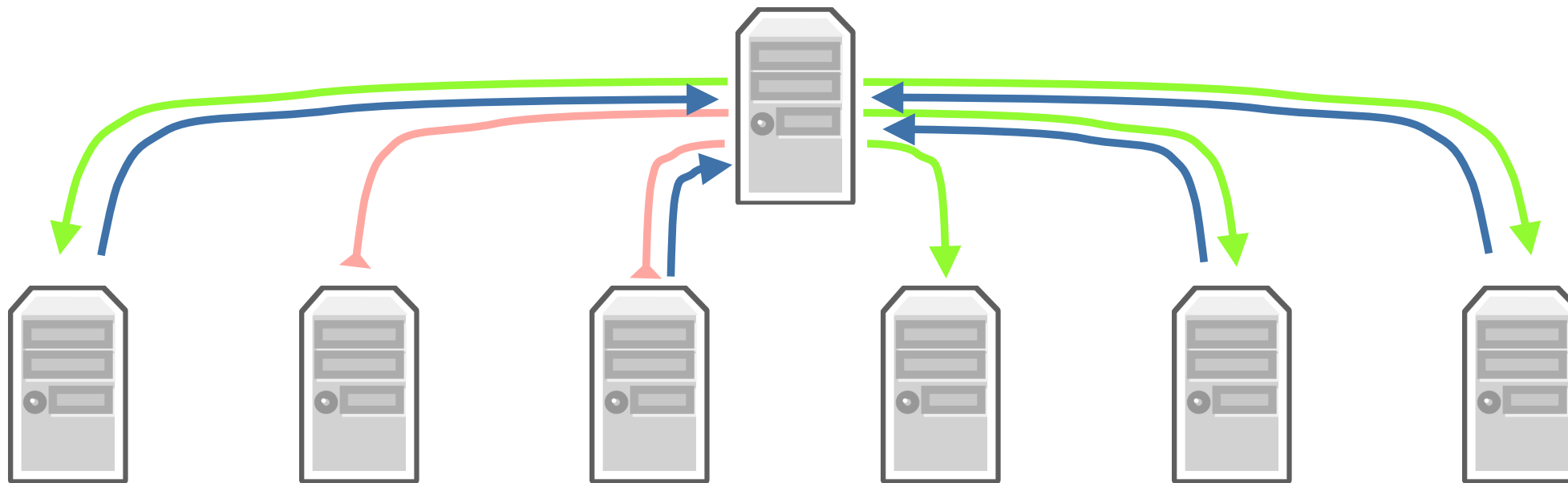
- BitTorrent breaks a file up into  $N$  pieces
  - ▶ For throughput, pieces are large: 256kB-1MB
  - ▶ For latency, broken into subpieces
- Hashes of pieces in torrent provide end-to-end integrity
  - ▶ Hash computes a short summary of a piece
  - ▶ Cryptographically strong hashes: hard to create a piece of data that has a particular hash (more in security lectures)
  - ▶ HBO's Rome series: blacklisting peers

# What to Say?

- Peers exchange metadata on what pieces they have
- Download rarest pieces: *rarest first* policy
- When down to the last few pieces, ask for them from multiple peers

# Whom To Talk To?

- Use *Tit-for-Tat* (TFT) policy: upload data to peers that give you data
- Most peers are “choked” and get no data
- Order unchoked peers by download rate, choke all but  $P$  best (e.g., 4,  $\sqrt{C}$ )
- Occasionally unchoke a random peer (might find way into  $P$  best)



# BitTyrant

[http://www.usenix.org/events/nsdi07/tech/piatek/piatek\\_html/bittyrant.html](http://www.usenix.org/events/nsdi07/tech/piatek/piatek_html/bittyrant.html)

- Can you game the BitTorrent Tit-for-Tat system?
- Many peers give more than they take
  - ▶ Give a peer just enough that it unchokes you
  - ▶ Convince as many peers as possible to unchoke you
  - ▶ Share capacity across more peers rather than give each peer more
- Leads to a 70% median performance gain!

# BitTorrent Summary

- Torrent file (.torrent) describes file to download
- File broken into pieces, each with a SHA1 hash
- Client finds peers through a tracker or DHT
- Clients connect over TCP/IP
- Clients exchange metadata on what pieces they have
- Clients try to download *rarest-piece-first*
- Clients “choke” most peers, send data to  $P$  best peers: *tit-for-tat*