

BitTorrent

A very popular Peer-to-Peer (P2P) system used today is BitTorrent.

BitTorrent itself is actually a set of protocols. Programs that implement the protocols are known as peers rather than as clients or servers. In fact, peers have both client and server capabilities.

[Bittorrent, Explained \(video by Andi Kola\)](#)

Please watch this video before continuing!

The [BitTorrent Protocol Specification](#) was invented by Bram Cohen

[Bittorrent Protocol Specification v1.0](#) is an unofficial specification designed to clarify and enhance Cohen's specification

uTorrent transport protocol is a BitTorrent transport protocol over UDP that regulates BitTorrent traffic

A gentle narrative introduction to creating a BitTorrent client was written by Kristen Widman; see [How to Write a Bittorrent Client, Part 1](#) and [How to Write a Bittorrent Client – Part 2](#)

Metainfo (Torrent) files

The file, or files, that one wishes to obtain using a BitTorrent client is/are known as the torrent

A file describing the torrent is known as the "metainfo" file; it is more commonly referred to as the torrent file because its name ends with the .torrent extension

The metainfo file consists of two primary key value pairs:

- announce -- the URL of the tracker
- info -- a dictionary that contains metadata about the torrent

In the case of a single file, the info dictionary consists of:

- name -- the name of the file (purely advisory)
- piece length -- the length of each piece of the file; all pieces are the same size except for possibly the last piece
- pieces -- a string that comprises the concatenation of the SHA1 hashes of all of the pieces
- length -- the total length of the file in bytes

Data is stored in the metainfo file using a pseudo-binary encoding standard, referred to as "[bencoding](#)". It is not truly binary because the sizes are recorded in simple decimal format.

Types of data encoding:

- Byte strings
- Integers
- Lists
- Dictionaries

Tracker requests

Once it has retrieved a metainfo file, a BitTorrent client may request information about the peers that have pieces by making a GET HTTP request to the tracker

Peer Wire Protocol

Armed with a list of IP addresses of peers, a client connects to another client and begins with a handshake

Handshake:

<length of protocol name string as a single byte><protocol name (string)>
(*cont.*) <eight (8) reserved bytes><info hash><peer id>

Once the handshake is completed, messages are exchanged that direct further activity.

Messages are fixed length.

Each message begins with its length as a four-byte big-endian value.

The message ID is a single byte.

Within messages, integers are encoded as four-byte big-endian values.