Raymond Sun
Henry Xiang hwx1

The MessageHandler class encapsulates the incoming and outgoing behavior of the connection socket. It can be passed messages to send to remote peers, and waited on for incoming messages. MessageHandler will automatically process incoming messages into a ByteBuffer of the appropriate length with the message ID as the leading byte.

The BTData class allows data about the RUBTClient to be serialized, such as amount uploaded and amount downloaded.

TrackerMessage decodes the tracker data from the .torrent file.

Peer simply holds information about the peers such as IP, port, and generated ID. It also holds an arraylist of integers to keep track of the pieces that each peer holds.

The program spawns a thread for listening to a ServerSocket to establish connections, as well as a thread for establishing handshakes. It then goes through the list of peers in the tracker and attempts to connect to valid peers. The program then goes through the regular BitTorrent protocol exchange. If the user ends the program, all the pieces currently downloaded will be saved to an incomplete file, and download will resume upon startup the following attempt.

The NumPieces class creates an object with a field for storing the index for a piece, as well as a field for how many current peers have that piece. We use this class to help find the rarest pieces among the peers. Our rarest piece algorithm utilizes an array of NumPieces objects in conjunction with a peer list, in order to find the number of times each piece shows up in each peer's bitfields. Through this we can determine the rarest pieces.