

Alexander DeOliveira
Brendan Parks

The main class is **RUBTClient.java**. This class parses the torrent file passes the information into TorrentInfo and creates the Tracker and Peer objects.

The **Tracker** object which is created by **RUBTClient** will talk to the tracker and retrieve and decode the response finding the peer list. After returning the proper peer back to **RUBTClient**.

A **Peer** object is created using the IP from the **Tracker** object. The **Peer** class will perform a handshake with the peer, receive a bitfield message from the peer, send an interest message, receive unchoke message, then request pieces and download pieces. All of the pieces found in the **Peer** class are sent back to **RUBTClient** and are written to the file whos name is the second parameter.

The **Message** class is a utility class, to be used in the future to fully decode all messages.

Caveats: We had a fantastic (read: miserable) time debugging problems involving character encodings. We eventually learned that it is best to just let the bytes be themselves.

Lessons Learned: Follow protocols “to the T”, or else you will pay for it later!