BitTorrent Help Session

Brown University CS 1680

2016-12-01

Project 4: BitTorrent Client

- Minimum Requirements
  - Can Download 1 multi-file torrents from multiple peers
  - Can Seed 1 multi-file Torrent to multiple peers
  - Allow viewing statistics (such as bytes downloaded, seeders/leechers)
  - *See handout for more details*

- Suggested Implementations
  - Support multiple concurrent torrents
  - Fancy Choking or piece selection algorithms

Project 4: BitTorrent Client

- 2 person project
  - Lots of ways to break up work, example:
  - 1 person can do Piece, Bitmap, File related stuff
  - 1 person can do Peer setup, teardown, and protocol stuff
- Can work alone, but requirements remain the same

Project 4: BitTorrent Client

- DO's
  - read the specs: https://wiki.theory.org/BitTorrentSpecification
  - read the handout:
  - use Internet Archive https://archive.org/details/bittorrent
  - use Wireshark
  - use uTorrent, transmission, etc
  - Use whatever language you want

- DONT's
  - Copy code from github – there is a lot of free BitTorrent code online
  - Use illegal Torrents

Project 4: BitTorrent Client

- AWS EC2 instances will be used for your client
  - because otherwise you can’t accept incoming connections
  - NAT traversal can be implemented on your own if you feel like it

- Ports 6881-6889 will be opened up to accept on
  - advertise this port to the Tracker, so other peers can connect
  - 1 port per BitTorrent client, so you need to multiplex if hosting >1 torrents

Project 4: BitTorrent Client

- Support Stuff
  - Reference clients for windows/linux/Darwin in the pub directory
    - useful for debugging or a reference for what you are trying to do
    - no need to implement the UI this way – see transmission-cli
  - Some .torrent files from the Internet Archive
    - used by us for testing, you can use as an example
    - You’ll want to grab your own .torrent files from the Internet Archive though…

Project 4: BitTorrent Client

- Suggested Development Flow
  - Use your TCP/IP github repo and start a new branch off the 1st commit
    - or create a new repo
  - Start a client instance on your EC2 instance, serving your torrent
    - transmission-cli transmission/ref_torrents/Hamlet_334_archive.torrent -P 6881
    - btclient –debug=1 –port=6881 (for debugging your client)
  - Develop on your laptop for the Tracker connection, and torrent downloading
    - You can use Wireshark this way – very useful for initial debugging!
    - compare your requests to btclient or uTorrent or transmission or whatever
  - Develop on the aws instance for torrent seeding
    - cuz you can’t seed from your laptop
    - this is where your remote git repo will make development much faster

Project 4: BitTorrent Client

- Immediate Actions
    - Decide if your going to work with a partner or not
    - Email the TA's, asking for your VM
    - setup your github repo
    - download wireshark, uTorrent, transmission-cli and poke around
- Implement ALL Tracker/MetainfoFile stuff first
- Then Downloading
- Then Seeding

Project 4: BitTorrent Client

- Example & Questions