Creation and Distribution of a Torrent File in a Peer-to-Peer (P2P) Environment

NAME: NAVINESHARAN S

ROLL NUMBER: 2023115015

---

1. Introduction

Peer-to-Peer (P2P) computing is a decentralized communication model where each participant (peer) can act as both a client and a server. BitTorrent is a popular P2P protocol used for efficient file sharing without relying on a central server.

In this experiment, a .torrent file is created using **qBittorrent**, and the file is shared between two peers to demonstrate decentralized file distribution, seeding, downloading, and file integrity verification.

---

2. Objective

- To create a .torrent file for a selected file
- To configure tracker information
- To seed the file from one peer
- To download the file from another peer
- To verify file integrity after download

---

3. Software and Environment

Component	Details
Torrent Client	qBittorrent
Peer 1	Seeder (Original file owner)
Peer 2	Downloader
Tracker	Public UDP Trackers

Component	Details
File Shared	B.Tech_IT.pdf
Network	Internet / LAN

---

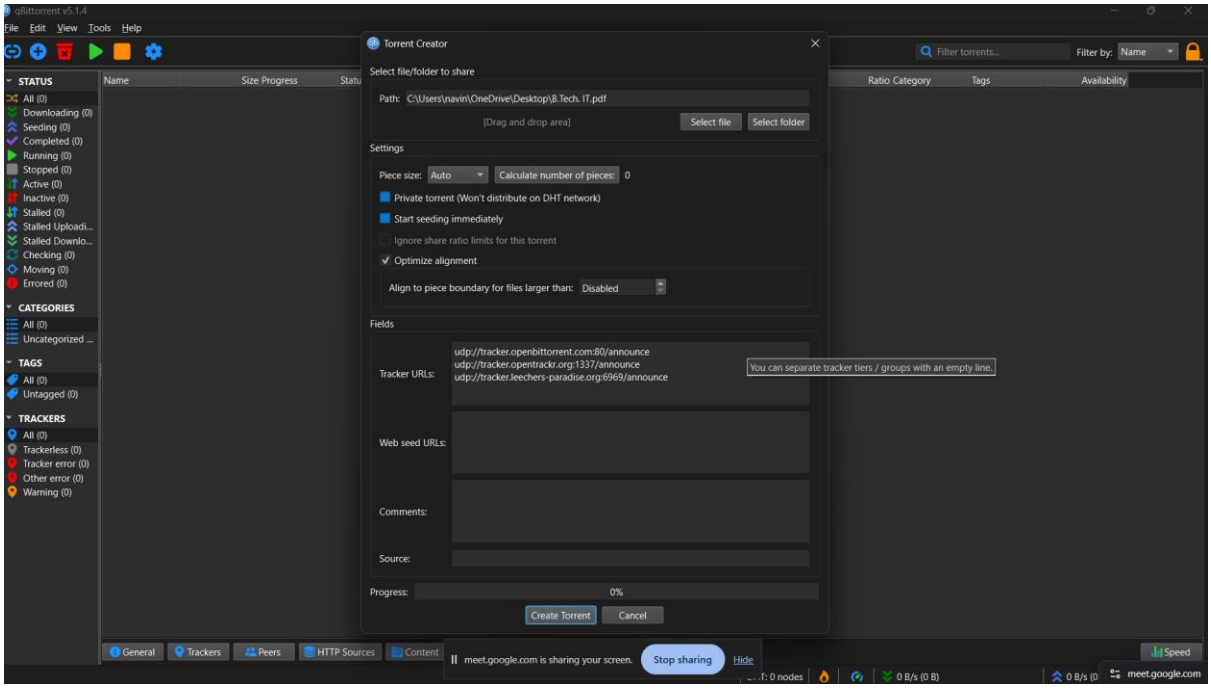
4. P2P Architecture

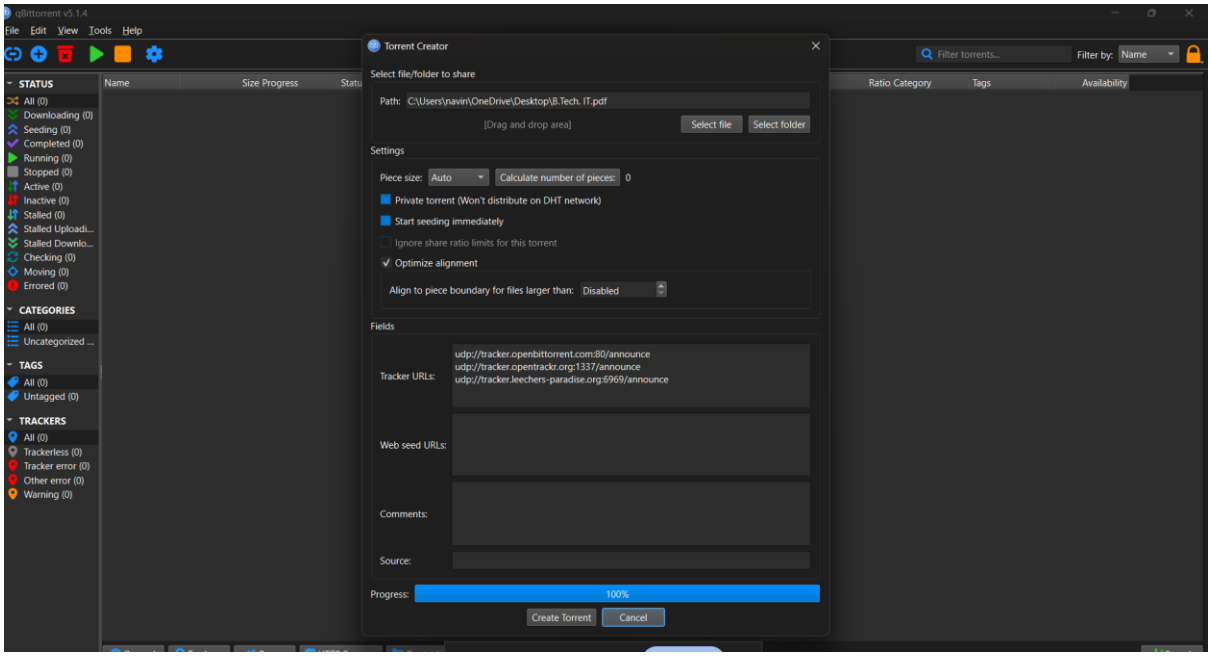
Peer 1 (Seeder) Peer 2 (Downloader)

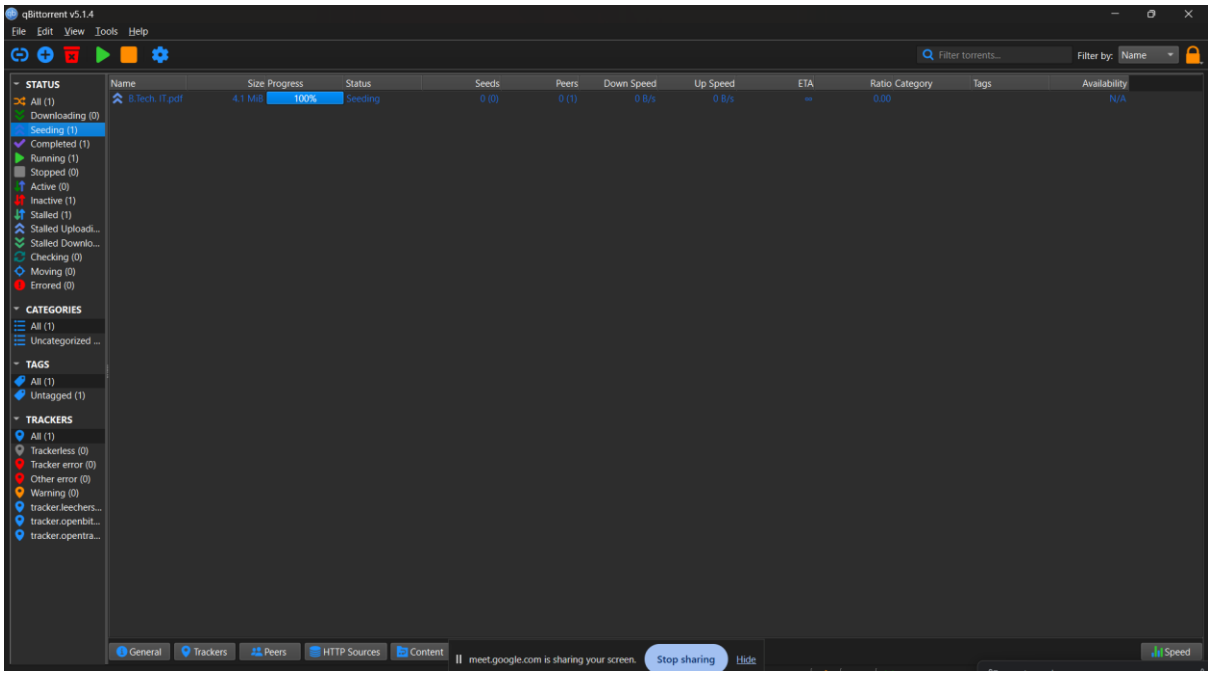
- No central server stores the file
- Peers exchange file pieces directly
- Tracker only helps peers discover each other

---

5. Torrent Creation Process (Peer 1 – Seeder)







Steps performed:

1. Opened **qBittorrent → Tools → Torrent Creator**
2. Selected the file B.Tech_IT.pdf
3. Added public tracker URLs:
4. udp://tracker.openbittorrent.com:80/announce
5. udp://tracker.opentrackr.org:1337/announce
6. udp://tracker.leechers-paradise.org:6969/announce

7. Clicked **Create Torrent**
8. .torrent file generated successfully
9. Torrent started **seeding** (100% progress)

---

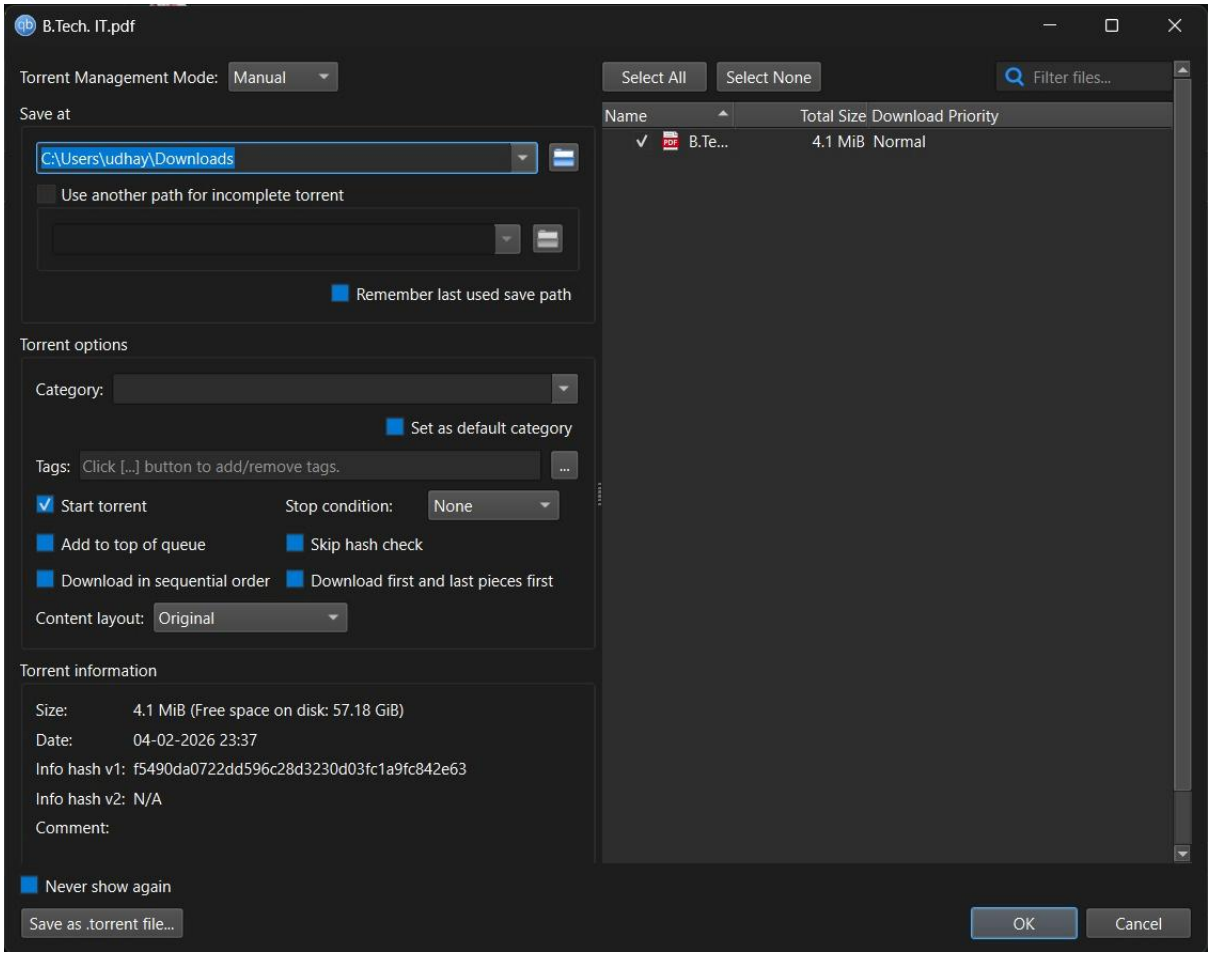
6. Tracker Configuration

Trackers are used to help peers find each other.

Tracker Type	Purpose
UDP Public Trackers	Peer discovery
DHT Network	Decentralized peer lookup
No central server	Fully decentralized sharing

---

7. Torrent Distribution Process (Peer 2 – Downloader)



Steps performed:

1. .torrent file shared with Peer 2
2. Peer 2 opened the torrent in qBittorrent
3. Selected download location
4. Download started from Peer 1 (seeder)
5. File downloaded completely
6. qBittorrent automatically verified file integrity (hash check)

---

8. File Integrity Verification

- BitTorrent divides files into pieces
- Each piece has a hash value
- After download, hashes are verified automatically
- Ensures **no corruption** during transfer

This confirms reliable P2P transfer.

---

9. Output Observed

Stage	Observation
Torrent creation	.torrent file generated
Seeding	100% seeding from Peer 1
Downloading	Peer 2 received file from Peer 1
Verification	File integrity verified successfully

---

10. Advantages of Torrent-Based P2P Sharing

- No central server required
- Efficient bandwidth usage
- Faster downloads using piece sharing
- High reliability due to hash verification
- Scalable with more peers

11. Conclusion

The torrent file was successfully created and distributed in a P2P environment using qBittorrent. The file was seeded from one peer and downloaded by another peer with complete integrity verification. This experiment demonstrates decentralized file sharing using BitTorrent concepts and highlights the effectiveness of peer-to-peer computing.