TD1-07 | Peer-to-Peer IPTV System over the Internet

Group Members
Wong Wai Keung (06616356)
Chan Siu Lung (06616021)

Project Supervisor: Dr. Danny H.K. Tsang
Introduction

In recent years, the Peer-to-peer (P2P) technology is one of the most influential and revolutionary technologies in the Internet development. This project aims to provide a peer-to-peer application into a video streaming system. It allows peers to watch video without delay and share their received data at the same time.

To implement the project, we tried to setup the server with random select address function to help peer to get video segments. As a result, it can reduce the workload of the server and improve the transmission of the system.

System Block Diagram

![System Block Diagram](image)

For the server side, it contains three main components:
1. A Network Manager
2. A Data Handler
3. Source Media file

For the peer side, it is run on a user’s computer. It contains five main components:
1. A Network Manager
2. A Data Handler
3. Video Segment
4. A Virtual Server
5. Media Player
Methodology

In order to get the streaming data for a peer joined the network, the server will assign two to three peers as its parent nodes to download video segments as shown in figure 2.

This assignment is based on random selection method by the server to distribute the loading of delivery evenly,

![Figure 2: Random parent nodes selection](image)

Video segments are treated as data blocks and each data block is 30k bytes long. Block sequence numbers are assigned starting from zero and then increase by one sequentially. The peer can request data blocks with specific sequence numbers from assigned parent nodes.

![Figure 3: Video segmentation](image)

Proposed User Interface

![Figure 4: Server Side Interface](image)  
![Figure 5: Peer Side Interface](image)
Results

Figure 6: Loading of server with increasing peers

Figure 6 shows the upload rate of server when increasing the number of peer. When more peers are connected, the workload in server side would not increase much because of data sharing between peers.

Conclusion & Further Development

To develop a Peer-to-peer feature in our system, we implemented the random assignment of peer node addresses in our server side. When peer connects, the server stored the information of this peer for future data exchange. In this stage, the peer side can receive the data segments and run the video smoothly in the media player after connecting to the server.

For further development, it can be modified the method in data exchange to perform the real time streaming and add a player list function on both sides since there is only one video that can be selected in this application now. It provides more choices for users.