

IT2 Beleg

Information

Author: Sebastian Mischke

[Aufgabenstellung](#)

Project

Bei dem Projekt handelt es sich um eine Client-Server-Anwendung, die mittels des [Real-Time-Streaming-Protokolls \(RTSP\)](#) eines Videostream überträgt. Dabei werden die eigentlichen Videodaten mittels des [Real-Time-Protokolls \(RTP\)](#) übertragen. Um den Einfluss von verlorengegangenen Paketen zu verringern, wird als Ausfallschutz eine [Forward-Error-Correction \(FEC\)](#) eingesetzt.

classdiagram

Server

Die Aufgabe des Servers ist es, aus den Frames einer `.mjpeg`-Datei sowohl RTP-Pakete als auch FEC-Pakete zu erzeugen und diese mittels eines Datagram-Sockets an den Client zu schicken.

Client

Der Client ist die grafische Bedienoberfläche des Nutzer. Er kann eine Verbindung mit dem Server aufbauen, diesem Befehle wie z.B. `Play` oder `Pause` schicken und empfangene JPEG-Bilder anzeigen. Zusätzlich zeigt er dem Nutzer Statistiken über die bisherigen Übertragungen an.

interface_client

RTPpacket

```
header[0] = (byte) ((Version << 6) | (Padding << 5) | (Extension << 4) | CC); // |VVPX|CCCC|
header[1] = (byte) ((Marker << 7) | (PayloadType & 0x7F)); // |M<--|PT->|
header[2] = (byte) (SequenceNumber >> 8); // SeqNum: Highbyte-Teil
header[3] = (byte) (SequenceNumber & 0x00FF); // Lowbyte-Teil
header[4] = (byte) (TimeStamp >> 24); // [xxxxxxxx|-----|-----|-----]
header[5] = (byte) ((TimeStamp >> 16) & 0x000000FF); // [-----|xxxxxxxx|-----|-----]
header[6] = (byte) ((TimeStamp >> 8) & 0x000000FF); // [-----|-----|xxxxxxxx|-----]
header[7] = (byte) (TimeStamp & 0x000000FF); // [-----|-----|-----|xxxxxxxx]
header[8] = (byte) (Ssrc >> 24); // Ssrc same as Timestamp
header[9] = (byte) ((Ssrc >> 16) & 0x000000FF);
header[10] = (byte) ((Ssrc >> 8) & 0x000000FF);
header[11] = (byte) (Ssrc & 0x000000FF);
```

FECpacket

Das FECpacket beinhaltet die Funktionen für die Forward-Error-Correction (FEC). Zum einen erlaubt es dem Server ein RTP-Paket zu erstellen, welches die Informationen eines FEC-Paketes besitzt, zum anderen bietet es dem Client die Möglichkeit, ein verloren gegangenes RTP-Paket wiederherzustellen.

Serverseite

Clientseite

```
int get_missing_nr() {
    int next = this.to_frame - this.FEC_group;
    for (int i = 0; i < rtp_nrs.size(); i++) {
        if (rtp_nrs.get(i) == next + 1) {
            next = rtp_nrs.get(i);
        } else {
            return next + 1;
        }
    }
    return this.to_frame;
}
```