Multimediakommunikation

Gedächtnisprotokoll vom 9. April 2019

Note: 1,0

1 Medien und Problematik bei der Übertragung

Frage: In welche beiden Kategorien haben wir Medien eingeteilt?
Anwort: In zeitdiskrete und zeitkontinuierliche Medien.
Frage: Was haben wir da für Beispiele kennen gelernt?
Anwort: Diskret: Bild, Text. Kontinuierlich: Video, Audio.
Frage: Welche Problematik kommt gerade bei der Übertrageung von kontinuierleichen Medien hinzu?
Anwort: Der Delay-Jitter.
Frage: Was ist der Delay-Jitter?
Anwort: Die Varianz in der Ende-zu-Ende Verzögerung.
Frage: Wie kann man das Kompensieren?
Anwort: Mit einem Playout Buffer auf Empängerseite.
Frage: Was ist das und was hat es für Nachteile?
Anwort: Ein Buffer, in dem ankommende Pakete vor der Wiedergabe gepuffert werden. Es erzeugt einen zusätzlichen Buffer Delay. Allerdings hat man keinen Rate-Jitter mehr bei der Wiedergabe.
Frage: Woher weiß der Empfänger, wann ein Paket gespielt werden soll?
Anwort: Anhand des Zeitstempels im Paketkopf.
Frage: Mit welcher Clock ist dieser Zeitstempel erzeugt worden?
Anwort: Mit der Sampling Clock des Absenders.
Frage: Zu was verhilft uns das?
Anwort: Zu intra-Stream Synchronisation.
Frage: Was kann man als Empfänger bei Paketverlust machen, um diese zu tolerieren?
Antwort: Bei Stimme: wiederholen des zuletzt empfangenen Paketes, da Stimme oft periodische Wellenform hat. Bei Audio bietet sich Interpolation an.
Frage: Was kann man auf Senderseite machen, um Paketverlust vorzubeugen?
Antwort: Man kann Vorwärtsfehlerkorrektur anwenden.
Frage: Was gibt es hierfür für Varianten?
Antwort: Generisch und verschränkt.
Frage: Was wird bei der verschränkten Variante gemacht?
Antwort: Dort werden beispielsweise über die Pakete 1, 5, 9, 13, ... eine Parität berechnet und diese gesendet.
Frage: Gegen was besonders eignet sich das?
Antwort: Gegen Burst-Losses der Länge L.

Frage: Was ist der Nachteil dabei?
Antwort: Der Overhead und man muss auf die ganze Gruppe von Paketen warten.

2 Analoge und digitale Signale

Frage: Was muss alles passieren, damit meine Stimme beim Empfänger zu hören ist?
Anwort: Zuerst muss es einen Stimulus geben, man muss sprechen. Danach nimmt ein Mikrofon das Signal auf und schickt es durch einen Tiefpass damit wir auch nur Frequenzen an den AD Wandler geben, die er auch korrekt samplen kann. Danach werden im AD Wandler zu diskreten Zeitpunkten Samples abgetastet. Diese werden dann einem Repräsentanten im Interval zugeordnet - quantisiert. Danach wird das Signal kodiert und in Pakete eingeteilt. Es folgt die Übertragung zum Empfänger. Dieser bröselt die Pakete wieder auf und dekodiert das Signal. Es wird schließlich über einen Lautsprecher wiedergegeben.
Frage: Wie lange darf das ganze dauern, um Interaktivität sicher zu stellen?
Anwort: Höchstens 400ms. Gewünscht ist aber alles unter 150ms.
Frage: Was passiert, wenn es länger als 400ms dauert?
Anwort: Dann weiß man nicht ob mein Gegenüber gerade spricht und ich rede ihm dadurch rein.
Frage: Was ist die übliche Abtastrate?
Anwort: 8000Hz.
Frage: Wenn man aber nun HD Qualität haben will, welche Abtastrate nimmt man denn da?
Anwort: 16000Hz.
Frage: Malen Sie mal ein Sinus Signal auf und beschriften Sie die Achsen. Was haben diese für Einheiten?
Anwort: Auf der X-Achse befindet sich die Zeit und auf der Y-Achse haben wir deziBel.
Frage: Was gibt uns deziBel an?
Anwort: Den relativen Schalldruck des Signals zum Luftdruck.
Frage: Wenn wir nun Stimme quantisieren wollen, was für eine Anordnung der Quantisierungsintervalle bietet sich da an?
Anwort: Intervalle nach U-Law. Nahe der Amplitude haben wir eine hohe Auflösung und nach Außen hin nimmt die Größe der Intervalle exponentiell zu.
Frage: Warum macht man das?
Anwort: Weil nahe der Amplitude Störungen vom Menschen eher wahrgenommen werden.
Frage: Wie nennt man diese Störung?
Anwort: Quantisierungsfehler.

3 RTP

Frage: Wir hatten zur Übertragung von Streamdaten ein Protokoll kennen gelernt. RTP. Was bringt uns das?
Anwort: Es liefert uns Transport von Streamdaten. Mixer ermöglichen große Konferenzen und Translators können den Datenstrom verändern. Im Kopf des Paketes wird die Quelle des Stroms eindeutig identifiziert, Sequenznummern ermöglichen Erkennung von Paketverlust, der Zeitstempel ermöglicht uns Intra-Stream Synchronisation. Dann gibt es noch das Padding-Present Bit, Extension-Header-present-bit, die Versionsnummer, den Payload Typen und bis zu 15 identifizierbare Absender. Der Erzeuger das Streams wird über die Synchronization Source identifiziert.
Frage: Was sagt uns der Payload Typ?
Anwort: Das ist Profilabhängig. Bei RTP/AVP bestimmt es den für die Erzeugung des Payloads verwendeten Codec.
Frage: Wie wird die Synchronization Source besimmt?
Anwort: Per Zufall.
Frage: Diese muss eindeutig sein. Was passiert, wenn es Kollisionen gibt?
Anwort: Dann muss eine Source eine neue SSRC würfeln.
Frage: Wie erkennt man eine Kollision?
Anwort: Anhand der SSRC und verschiedenen Absenderadressen.

4 RTCP

Frage: Jetzt wollen wir auch noch ein bisschen Feedback haben. Dazu gibt es RTCP. Was bietet uns das?
Anwort: Hier gibt es Sender-Reports und Receiver-Reports. Diese enthalten einen Reception-Block, der Informationen über die Qualität des Streams gibt.
Frage: Was steht da so drin?
Anwort: Die relative und absolute Anzahl an Paketen, die nicht ankamen. Der geschätze Interarrival-Jitter, der vergangene Zeit zum letzen zu diesem Stream empfangenen Sender-Reports und dessen mittlere 32 bit des NTP Zeitstempels.
Frage: Was überträgt der Sender noch, bevor er Reception-Blocks sendet?
Anwort: Packet Count, Octet Count, einen NTP Zeitstempel und die gleiche Zeit nochmal, allerdings umgewandelt in die für den Stream über den berichtet wird RTP Zeit. Also derjenige NTP Zeitstempel, den ein Paket in dem Moment des NTP Zeitstempels hat.
Frage: Wozu wird das genutzt?
Anwort: Für Inter-Stream Synchronisation.

5 DASH

Frage: Nun gab es eher ein Trend dazu, über HTTP zu streamen. Da kennen wir DASH. Was ist daran besonders?
Anwort: Bei DASH werden die Streams in Perioden eingeteilt. Jede Periode hat verschiedene Adaption Sets, die die Daten in verschiedenen Bitraten enthalten. Diese Adaption Sets bestehen aus kleinen Segmenten.
Frage: Wie groß ist so ein Segment?
Anwort: Dateigröße ist vom Codec abhängig. Zeitlich gesehen, etwa 1s - 5s.
Frage: Wieso wählte man gerade diese Größen?
Anwort: Das ist ein Tradeoff zwischen Geschwindigkeit, in der ein Client auf Schwankungen in der Bitrate reagieren kann, und Dateianzahl auf dem Server und Overhead wegen der ganzen GET Anfragen, die der Client stellen muss.
Frage: Was ist noch besonders hinsichtlich des Transportprotokolls, das DASH verwendet?
Anwort: DASH verwendet TCP.
Frage: Warum gerade TCP?
Anwort: Wenn man es beispielsweise bei Video-on-Demand einsetzt, ist ein üblich eine gute Sekunde Playback-Delay zu haben. In dieser Zeit sollte TCP es schaffen, alle fehleneden Pakete zusammen zu haben. Bei Telefonaten ist das nicht geeignet. Man kann hier also von den Vorteilen von TCP profitieren.
Frage: Denken sie mal an die höheren Schichten, fällt Ihnen da noch ein Grund für TPC ein?
Anwort: Ja, bisher ist HTTP nur über TCP definiert. Wir sind also gezwungen es zu nutzen, wenn wir HTTP haben wollen.

6 SIP

Frage: Jetzt hatten wir noch über VoIP gesprochen. Was ist denn da das gängige Protokoll zur Signalisierung?
Anwort: SIP (yeah)
Frage: Aus welchen 3 Teilen besteht eine SIP Nachricht? *Antwort:* Kopf, Request Line und Body. *Frage:* Malen Sie mal zwei Clients, Alice und Bob auf. Bob hat einen Proxy. Alice will Bob anrufen, wie verläuft das?
Gegenfrage: Ist Bob bereits registriert?
Gegenantwort: Nein, und er muss sich erst authentifizieren.
Anwort: Bob sendet ein REGISTER an seinen Proxy. Dieser antwortet mit 183 PROXY AUTHENTICATION REQUIRED und schickt Bob eine Nonce. Diese wird mit seinem einzugebenen Passwort in einen Algorithmus geworfen und das Ergebnis X hängt Bob bei einem erneuten REGISTER an. Der Proxy überprüft, ob X zu der Nonce und dem auf dem Proxy gespeicherten Passwort von Bob passt. Falls ja, sendet er ein 200 OK. Bob ist dann registiert.
Nun kommt Alice ins Spiel. Damit sie weiß, wie sie Bobs Proxy erreichen kann, nutzt sie das DNS. Zuerst wird eine NAPTR Anfrage für SIP über TCP oder

UDP oder SIPS gestellt. Dadurch erfährt sie die unterstützten Protokolle und offenen Ports. Diese werden dann für den SIP Transport verwendet werden. Danach stellt sie eine SRV RR Anfrage an einen Server, der das gewünschte Protokoll unterstützt. Dadurch kommt sie an eine Liste von Load-Balancing Servern, die sie nutzen kann. Davon wählt man den mit der kleinsten Prioritätszahl und dem größten Gewicht. Zuletzt folgt die A Record Anfrage an diesen Server. Dadurch erhält sie die IP Adresse von Bobs Proxy. Dorthin schickt sie das INVITE.
Der Proxy schickt das INVITE an alle Geräte weiter, die Bob dort registriert hat. Er antwortet Alice mit 100 TRYING, was Sendewiederholungen von Alice verhindert. Die Geräte von Bob antworten mit 180 RINGING dem Proxy, welche er an Alice weiterleitet. Sie bekommt ein Tuten vorgespielt. Wird an einem Gerät abgenommen, so sendet es ein 200 OK an den Proxy, damit er den anderen Geräten per CANCEL Nachricht signalisieren kann, nicht weiter zu klingeln. Dieses wird mit einem 200 OK bestätigt und die noch offene INVITE Transaktion mit einem 487 REQUEST CANCELLED quittiert. Das 200 OK vom abgehobenen Gerät leitet der Proxy auch weiter an Alice, die dies Ende-zu-Ende Bob mit einem 200 OK beantwortet.
Frage: Woher kennt Alice Bobs IP Adresse?
Anwort: Diese steht im 200 OK, genauer, dort im Contact Field.