

# Strumieniowanie danych

## 1. Wstęp teoretyczny:

### Informacje ogólne:

Strumieniowanie jako pojęcie odnoszące się do transmisji danych powstało we wczesnych latach osiemdziesiątych natomiast dopiero pod koniec XX wieku rozpowszechnienie dostępu do sieci oraz zwiększenie jej przepustowości oraz mocy obliczeniowej procesorów sprawiły że można było mówić o efektywnym strumieniowaniu danych w praktyce przy użytkowaniu komputerów osobistych. Follansbee określa strumieniowanie danych jako „ciągły transfer danych z jednego komputera do drugiego w czasie rzeczywistym” przy użyciu pakietów danych. Największą różnicą pomiędzy strumieniowaniem a pobieraniem danych jest fakt że przy pobieraniu dane nie mogą zostać odtworzone dopóki pobieranie nie zostanie zakończone. Strumieniowanie pozwala natomiast na odtwarzanie danych w momencie ich odbierania.

Rozróżniamy dwa sposoby strumieniowania danych:

- a) na żądanie (on demand) – strumienie są dostępne na serwerze przez dłuższy czas i są gotowe do odtwarzania
- b) na żywo (live) – dane są dostępne tylko w konkretnym momencie tak jak w przypadku audycji radiowych. Nie ma możliwości przewijania odtwarzanej audycji.

### Rodzaje transmisji wideo:

- a) Transmisja jeden-do-jednego (ang. unicast), w których pojedynczy pakiet danych przesyłany jest od nadawcy do jednego odbiorcy, dokładnie pod jeden adres. Transmisja może być zrealizowana np. w oparciu o protokół RTP. Za pomocą tego protokołu klient kontaktuje się z serwerem. Serwer odpowiada klientowi poprzez RTP, przysyłając informację opisującą materiał wideo jako sesję strumieniowania. Sesja strumieniowania może składać się z jednego lub więcej strumieni, np. strumienia audio i strumienia wideo. Serwer przekazuje klientowi informację, jak dużo strumieni powinien on oczekiwać oraz podaje dokładne dane dotyczące m.in. typu przesyłanych danych i zastosowanego kodeka.
- b) Transmisja jeden-do-wielu (ang. multicast) składa się z pojedynczego pakietu danych, który adresowany jest do grupy odbiorców, przy czym ruter docelowy może wysyłać pakiety nie tylko do użytkowników końcowych, ale także do innych ruterów.
- c) Transmisje rozgłoszeniowe (ang. broadcast) składają się z pojedynczego pakietu danych, kopiowanego i przesyłanego do wszystkich węzłów sieciowych. Pakiet jest adresowany przez węzeł źródłowy specjalnym adresem rozsyłającym, a następnie przesyłany do sieci, która tworzy kopie pakietu i wysyła je do każdego węzła sieci.

## Formaty plików video:

Główne 3 formaty to: pliki AVI w środowisku Microsoft Windows, pliki MOV technologii Quick Time komputerów Apple oraz format MPEG, szczególnie wersja MPEG-4. (w bliskiej przyszłości także MPEG-7). Formaty te ulegają dalszej integracji. W plikach AVI i MOV wykorzystywane są te same algorytmy kompresji (te same kodeki), a kompresja MPEG-4 jest już traktowana jak kodek, dzięki czemu pliki zakodowane w MPEG-4 są plikami w formacie AVI.

W kamerach cyfrowych i kartach montażu wideo najważniejszy jest kodek DV, kompresujący obraz ze stałą kompresją 5:1, pozwalający w pliku 2GB zapamiętać 9 minut filmu. Film w formacie AVI z kodekiem DV (w skrócie w formacie DV) wymaga 3,7 MB transferu na sekundę. Kompresja DV pozwala zapamiętać film z wyśmienitą jakością, porównywalną z jakością analogową BetaCAM. Tak zakodowane filmy generują duże pliki wideo, które są znakomite do dalszego montażu cyfrowego, ale absolutnie nie nadają się do udostępniania i przesyłania w sieci Internet, ani nawet w sieciach lokalnych.

Format MPEG-4, zorientowany jest głównie na telewizję cyfrową, nie daje możliwości bezproblemowego przesyłania w sieciach, choć kompresja jest w tym przypadku bardzo efektywna. Kodowanie MPEG-4 czy też DivX, bazujący na idei MPEG-4, pozwala zapisać na zwykłej płycie CD (700 MB) standardowy film o długości 90 minut. Ale i w tym przypadku wymogi transmisji są bardzo wysokie – nawet 130 kbps. MPEG-4 obniża wymagania dotyczące szerokości pasma transmisji, ale odbywa się to kosztem bardziej złożonego algorytmu kompresji i dekompresji. Komputery muszą dysponować znacznie większą mocą obliczeniową. Pliki wideo w formacie MPEG-4 są znacznie bardziej wrażliwe niż MPEG-1 na utratę jakości obrazu, w przypadku zakłócenia transmisji pakietów, np. ich utraty.

## Protokoły transmisji danych:

Protokołem połączeniowym przesyłania danych w Internecie jest protokół TCP. Pozwala on niezawodnie przysłać dane, wykorzystując mechanizmy sekwencjonowania – potwierdzenia przesłania i retransmisji pakietów. W protokole TCP przeprowadza się efektywną kontrolę przepływu i przeciążeń oraz umożliwia pracę w pełnym układzie duplexowym i multipleksowanie. Protokół TCP nie obsługuje transmisji jeden-do-wielu. Protokołem, który obsługuje takie transmisje jest protokół UDP (ang. User Datagram Protocol). Protokół UDP jest bezpołączeniowym protokołem transportowym, który nie zapewnia żadnych gwarancji dostarczenia, gdyż nie posiada mechanizmów zabezpieczeń przed przesyłaniem zduplikowanych pakietów UDP, tzw. datagramów, nie kontroluje przepływu pakietów i nie retransmituje pakietów. Dzięki prostocie UDP możliwe natomiast stało się m.in. zredukowanie wielkości nagłówka protokołu, minimalizacja opóźnień w dostarczaniu danych, oprogramowanie dodatkowych funkcjonalności i oczywiście prowadzenie transmisji jeden-do-wielu. Z tych powodów protokół UDP nadaje się doskonale do strumieniowania plików wideo, gdyż pozwala na przesyłanie danych w czasie rzeczywistym, a ewentualne mechanizmy kontrolne mogą zostać zaimplementowane w protokołach wyższych warstw transmisji. UDP jest protokołem bezpołączeniowym, gdyż może wysyłać datagramy bez

nawiązania połączenia z odbiorcą, a droga ich przesyłania może zmieniać się podczas transmisji. W transmisji opartej na strumieniowaniu protokół TCP zastąpiono więc prostszym protokołem UDP, którego zadaniem jest jak najszybsze wysłanie pod wskazany adres strumienia pakietów. Protokół UDP nie pozwala przekazywać wystarczających informacji, potrzebnych odbiorcy strumienia wideo. Jednakże takie parametry - konieczne dla poprawnego odbioru wideo są przenoszone przez protokół RTP. Protokół do transportu danych aplikacji czasu rzeczywistego RTP stosuje specjalny mechanizm, aby stało się możliwe przesyłanie danych czasu rzeczywistego, takich jak wideokonferencje w sieciach bazujących na IP. W pierwszym kroku, strumień danych jest dzielony na pakiety, które transmitowane są oddzielnie. Transmisje mogą mieć charakter jeden-do-jednego lub jeden-do-wielu. Następnie po stronie odbiorcy, otrzymane pakiety muszą być ustawione w poprawnej kolejności. Są również synchronizowane z innymi napływającym strumieniami. Odbiorca otrzymuje również dodatkowe informacje zwane metadanymi. Większość działań RTP polega na oznakowaniu właściwych danych audio/wideo i ich źródła, przydziale numerów sekwencyjnych i stempli czasowych dla każdego z pakietów, a także na monitorowaniu warunków sieciowych oraz na transmitowaniu informacji o sesjach użytkowników. Sam RTP nie dostarcza metadanych oraz funkcjonalności monitoringu. Jest to realizowane z udziałem wspomagającego go protokołu kontroli RTCP – Real-time Transport Control Protocol. Protokoły UDP i RTP wzajemnie się uzupełniają. Zawarta w nagłówku UDP informacja o numerze portu, służy za identyfikator strumienia RTP pomiędzy nadawcą a odbiorcą. Transmisja audio stanowi odrębną sesję w stosunku do wideo. Obie transmisje są jednokierunkowe, pakiety kierowane są tylko od nadawcy do odbiorcy, nie ma pakietów zwrotnych, a ewentualna transmisja w przeciwnym kierunku, co ma miejsce np. w wideokonferencjach, będzie oznaczać kolejne sesje RTP. Transmisja zwrotna jest przydatna, jeśli nadawca pragnie otrzymywać informacje o jakości odbioru, aby móc ją kontrolować. W tym celu dla każdej sesji UDP/RTP nawiązywana jest osobna sesja kontrolna, identyfikowana przez własny numer portu UDP wyższy o 1 od numeru sesji monitorowanej, którą płyną informacje, zarówno od odbiorcy do nadawcy, jak i od nadawcy do odbiorcy. Strumień danych audio każdego z uczestników jest enkapsulowany w pakiety RTP o stałej długości, które następnie wysyłane są na uzgodniony adres i rozsyłane do wszystkich uczestników. Dzięki dodatkowej informacji zawartej w pakiecie odbiorca może skorygować prawidłową kolejność pakietów, zsynchronizować je, zidentyfikować format odtwarzanych danych i połączyć strumień danych w celu ich odtworzenia. Każde źródło identyfikowane jest poprzez unikatowy znacznik SSRC (ang. synchronization source identifier). SSRC jest identyfikatorem źródła synchronizacji, zawartym w nagłówku RTP. Protokół RTP/RTCP jest obecnie najczęściej stosowanym protokołem strumieniowania transmisji, stosowanym w Internecie. M.in. jest on wykorzystywany przez serwery RealServer oraz przez systemy telekonferencyjne, zbudowane zgodnie ze standardem H.323, takie jak np. Microsoft Netmeeting. Z kolei protokół RTSP – Real-time Streaming Protocol, przyjęty w październiku 1996 roku, jest protokołem aplikacyjnym strumieniowej transmisji, który posiada także mechanizmy sterowania multimedialną sesją: m.in. ma możliwość alokacji zasobów oraz rozpoczęcia sesji RTSP, rozpoczęcia odtwarzania, zawieszenia odtwarzania bez zwalniania zasobów, a także ma możliwość zwolnienia zasobów i zamknięcia sesji.

**Literatura:**

John G. Apostolopoulos, Wai- tian Tan, Susie J. Wee - „Video Streaming: Concepts, algorithms, and Systems”

K.Choroś - „Strumieniowanie plików wideo w systemach nauczania na odległość”

**Użyte oprogramowanie:**

**VLC media player** - odtwarzacz multimedialny rozwijany przez VideoLAN, rozpowszechniany na licencji GPL, może być także używany jako serwer w sieciach o dużym przesyśle. Może odtwarzać wiele rodzajów plików bez konieczności instalacji dodatkowych kodeków.

- Obsługiwane formaty: Ogg, MPEG, DivX, MP3 i inne.
- Oficjalna strona programu : <http://www.videolan.org/vlc/>
- Video LAN wiki: [http://wiki.videolan.org/Main\\_Page](http://wiki.videolan.org/Main_Page)
- Video LAN dokumentacja: <http://www.videolan.org/doc/>
- Strumieniowanie VLC - [http://wiki.videolan.org/Documentation:Streaming\\_HowTo](http://wiki.videolan.org/Documentation:Streaming_HowTo)

**Strumieniowanie (na serwerze):**

- Plik > Strumieniuj > w zakładce plik, wybór pliku, "Dodaj", wybór pliku do strumieniowania
- Strumieniuj (na dole) > "Następny" > "Cele, nowe cel" (ustawić protokół) > Dodaj > ustawić port i ewentualnie adres
- W "Opcje transkodowania" ustawić profil, wybrać odpowiednie kodeki.
- Strumieniuj (na dole)

**Odbieranie strumienia (na klientach)**

- Plik > Otwórz strumień z sieci > Zakładka "sieć", podaj adres URL sieci > **np.** <http://10.0.2.139:8080> lub "udp://@224.0.0.1:1234"
- Odtwarzaj

**Pliki filmowe do pobrania na stronie:** <http://heavy.metal.agh.edu.pl/>

## Scenariusz 1:

### Potrzebne:

- 2 lub 3 komputery PC z zainstalowanym programem VLC (CentOS)

### Przebieg ćwiczenia:

- Sprawdzić czy jest zainstalowany program VLC,
- Utworzyć na pulpicie nowy folder, w którym będą znajdować się pliki video, oraz zrzuty ekranu potrzebne do sprawozdania

1. Prowadzący zajęcia wyznacza wyjścia, profile i pliki które zostaną poddane testom.
2. Wyjścia – protokoły (port)
  - a. HTTP (:8080)
  - b. RTP (:5004) adres 224.0.0.1 oraz nazwa strumienia, np. /nazwa.sdp
  - c. UDP (:1234) adres 224.0.0.1 (ewentualnie adres IP klienta, który będzie odbierał strumień)
3. Profile strumienia.
  - a. H.264 + AAC (TS)
  - b. Theora + Vorbis (OGG)
  - c. Theora + Flac (OGG)
  - d. DIV3 + MP3 (ASF)
  - e. własny profil
4. Testowe pliki wideo: .....

**Pliki do pobrania na stronie:** <http://heavy.metal.agh.edu.pl/>

5. Podczas każdego z testów badamy parametry na komputerze działającym jako serwer oraz na jednym z klientów. Pozostali klienci działają (generują obciążenie) ale nie zbieramy z nich danych.

### Badamy:

#### Na serwerze:

Po rozpoczęciu strumieniowania notujemy: (Ctrl+J, zakładka szczegóły kodeka):

- a. nazwę pliku, oryginalny kodek i parametry pliku
- b. protokół wykorzystany do strumieniowania wraz z parametrami
- c. profil (enkapsulacja, kodek video, kodek audio) - przy definicji strumienia: "opcje transkodowania" -> "edytuj wybrany profil"
- d. za pomocą polecenia top w konsoli monitorujemy proces aplikacji VLC dla zużycia procesora (% CPU)

Przed wykonaniem zrzutu ekranu należy:

- a. zaznaczyć odpowiedni proces na liście procesów
- b. zatrzymać i ponownie uruchomić monitorowanie, aby zresetować statystyki
- c. uruchomić strumieniowanie na serwerze
- d. podłączyć wszystkich klientów do serwera

#### **Na kliencie:**

(notujemy parametry tylko jednego klienta, po podłączeniu do serwera wszystkich klientów).

Notujemy (informację o zastosowanym kodeku Ctrl+J):

- a. nazwę pliku
- b. kodek video
- c. rozdzielczość video
- d. informację o strumieniu audio

Po upływie określonego czasu transmisji około (np 15 sekund) wykonujemy zrzut ekranu na którym widać: (konfiguracja identycznie jak dla serwera):

- a. obciążenie procesora
- b. zakładkę statystyki (Ctrl+J)
  - a) przepływowość wejściowa (kb/s)
  - b) przepływowość strumienia (kb/s)

Dodatkowo notujemy subiektywną jakość obrazu (opóźnienia, artefakty, czas buforowania, jakość obrazu)

6. Testy wykonujemy dla wszystkich możliwych kombinacji elementów (protokołów, profili, materiałów video) wybranych przez prowadzącego.

7. W wypadku nieprawidłowych proporcji wyświetlanego obrazu, zmieniamy go w kliencie przez „Obraz>stosunek rozmiarów>16:10”

8. Na końcu pracy **!!!USUWAMY!!!** folder w którym przechowywane były strumieniowane materiały video oraz zrzuty ekranu wykonane podczas laboratorium.

Przykładowy wpis w tabeli notatek dla jednej serii testów (+ 1 zrzut ekranu z serwera i 2 z klienta ):

przykładowy wpis w tabeli notatek dla jednej serii testów (+ 1 zrzut ekranu z serwera i 2 z klienta )

| protokół     | Parametry strumienia                                                                                                                             | Nazwa i parametry pliku                                                                         | Serwer                                                                                                                                                                  | Klient                                                                                                                                                                                       |
|--------------|--------------------------------------------------------------------------------------------------------------------------------------------------|-------------------------------------------------------------------------------------------------|-------------------------------------------------------------------------------------------------------------------------------------------------------------------------|----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| HTTP (:8080) | Video-H.264 + AAC (MP4); enkapsulacja MPEG-TS; kodek obrazu H-264, przepływowość 300kb/s fps 60; kodek dźwięku: flac, 128kb/s 2 kanały, 441000Hz | Transformers. mov, kodek video: H264-mpeg4-AVC, 852x354, kodek audio: MPEG AAC, 48kHz, 1536kb/s | Ilość klientów: 2<br>Wykorzystanie sieci: (wysłane): 3 344B/sec<br>Średnie zużycie procesora: 26%,<br>Ilość wątków: 4<br>Uwagi: chwilowe utraty płynności pracy systemu | Wykorzystanie sieci: (odebrane): 721 B/s;<br>Średnie zużycie CPU: 8%;<br>przepływowość wejściowa: 7719kb/s<br>przepływowość strumienia: 9475 kb/s<br>Jakość zadowalająca: rzadkie artefakty. |
|              |                                                                                                                                                  |                                                                                                 |                                                                                                                                                                         |                                                                                                                                                                                              |
|              |                                                                                                                                                  |                                                                                                 |                                                                                                                                                                         |                                                                                                                                                                                              |
|              |                                                                                                                                                  |                                                                                                 |                                                                                                                                                                         |                                                                                                                                                                                              |

Powyższe notatki należy opracować w osobnych tabelach i opracować wg poniższych wskazówek.

**Na koniec usuwamy wszystkie wygenerowane zrzuty ekranu, oraz ściągnięte pliki !!**

### Wyniki pomiarów:

a. na podstawie wyników opracować statystycznie i porównać (przy zachowaniu innych jednakowych parametrów)

- zużycie zasobów klienta i serwera (procesor),
- jakość obrazu (artefakty, ilość zgubionych klatek)

b. w zależności od

- ilości klientów
- zastosowanego protokołu,
- parametrów kodeka,
- rozdzielczości materiału video

c. wykadrować, pomniejszyć i wydrukować odpowiednie zrzuty ekranu (każdy protokół, kodek, materiał video)

d. opisać spostrzeżenia, wyciągnąć wnioski