

Zarządzanie ruchem i jakością usług w sieciach komputerowych

Część 1 wykładu

SKO2

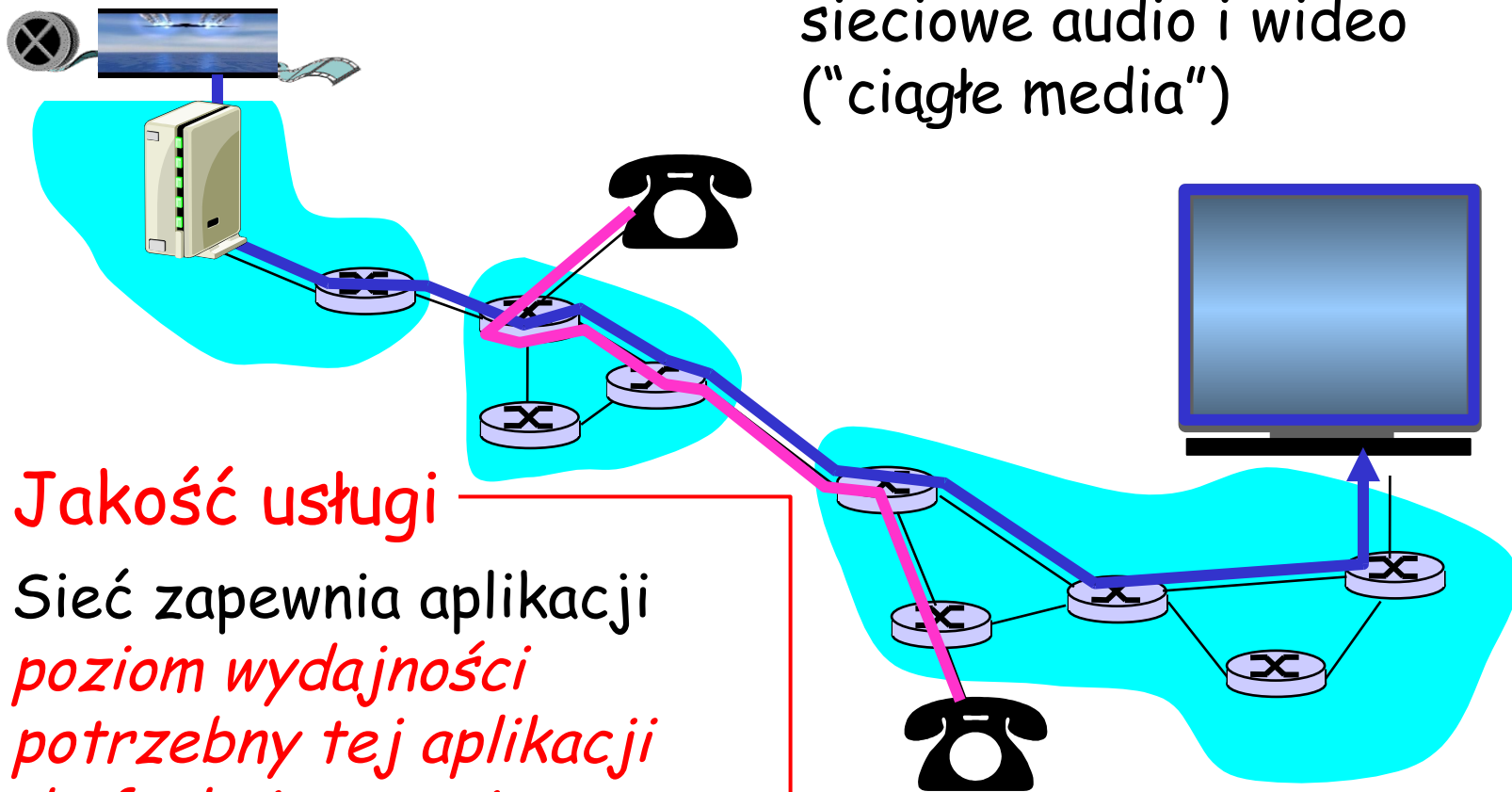
Mapa wykładu

- ❑ Wprowadzenie
 - 10 trendów rozwoju sieci
- ❑ **Komunikacja multimedialna w sieciach IP**
- ❑ Techniki QoS
 - ATM
 - IEEE 802.1D
 - Integrated Services i Differentiated Services
 - MPLS
- ❑ Problemy i perspektywy rozwoju tych technologii

Multimedia, jakość usług: co to jest?



Aplikacje multimedialne:
sieciowe audio i wideo
("ciągłe media")



Jakość usługi

Sieć zapewnia aplikacji
poziom wydajności
potrzebny tej aplikacji
do funkcjonowania.

Cele wykładu o komunikacji multimedialnej w sieciach IP

Zasady

- ❑ Klasyfikacja aplikacji multimedialnych
- ❑ Identyfikacja usług sieciowych, których potrzebują aplikacje
- ❑ Możliwie najlepsze wykorzystanie usługi best-effort

Protokoły i architektury

- ❑ Specjalne protokoły dla best-effort

Zarys wykładu o komunikacji multimedialnej w sieciach IP

- Aplikacje sieci z jakością usług
- Przesyłanie strumieniowe przechowywanych plików audio i wideo
 - RTSP
- Multimedia czasu rzeczywistego: studium przypadku telefonii internetowej
- Protokoły dla interaktywnych aplikacji czasu rzeczywistego
 - RTP, RTCP
 - SIP
- Poza best-effort
- Mechanizmy szeregowania i kontroli

Aplikacje sieci z jakością usług



Klasy aplikacji multimedialnych:

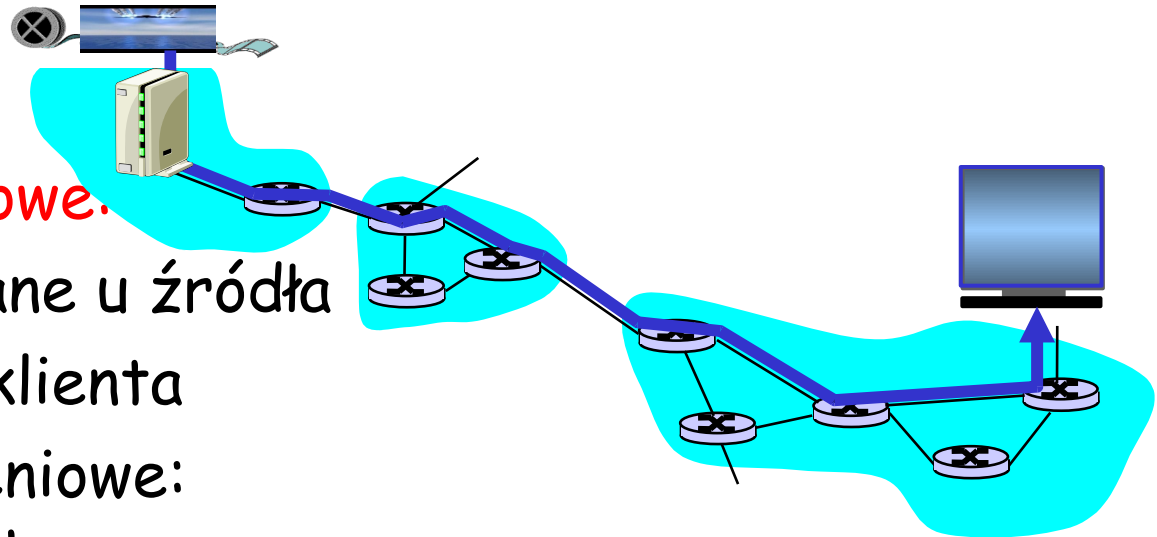
- 1) Przesyłanie strumieniowe przechowywanych plików audio i wideo
- 2) Przesyłanie strumieniowe dźwięku i obrazu "na żywo"
- 3) Interaktywny dźwięk i obraz w czasie rzeczywistym

Rozsynchronizowanie (ang. *jitter*) jest to zmienność opóźnień pakietów w pojedynczym strumieniu pakietów

Podstawowe charakterystyki:

- ❑ Na ogół **podatne na opóźnienia**
 - Opóźnienie zupełne
 - Zmienność opóźnienia
- ❑ ale **odporne na straty**: rzadko występujące straty powodują drobne zakłócenia
- ❑ Antyteza komunikacji niezawodnej, która nie jest odporna na utratę, ale na opóźnienia - tak.

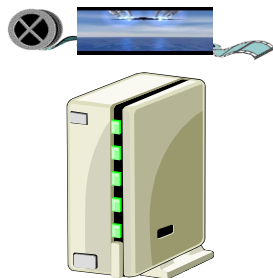
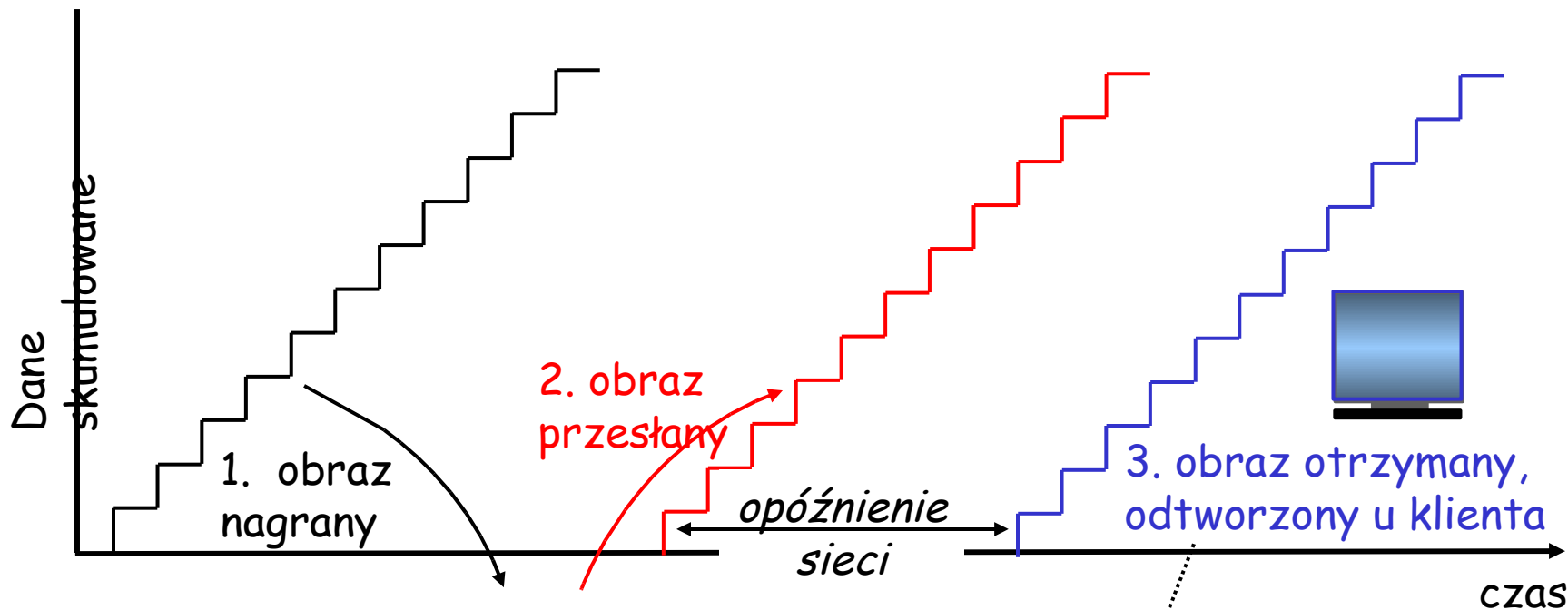
Przesyłanie strumieniowe przechowywanych multimediiów



Przesyłanie strumieniowe.

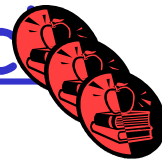
- ❑ media przechowywane u źródła
- ❑ Transmitowane do klienta
- ❑ przesłanie strumieniowe:
 - odtworzenie u klienta
 - rozpoczyna się *przed* otrzymaniem wszystkich danych
 - ❑ Ograniczenie czasowe dla danych pozostałych do przesłania: w czasie wystarczającym na odtworzenie

Przesyłanie strumieniowe przechowywanych multimediiów: Co to jest?

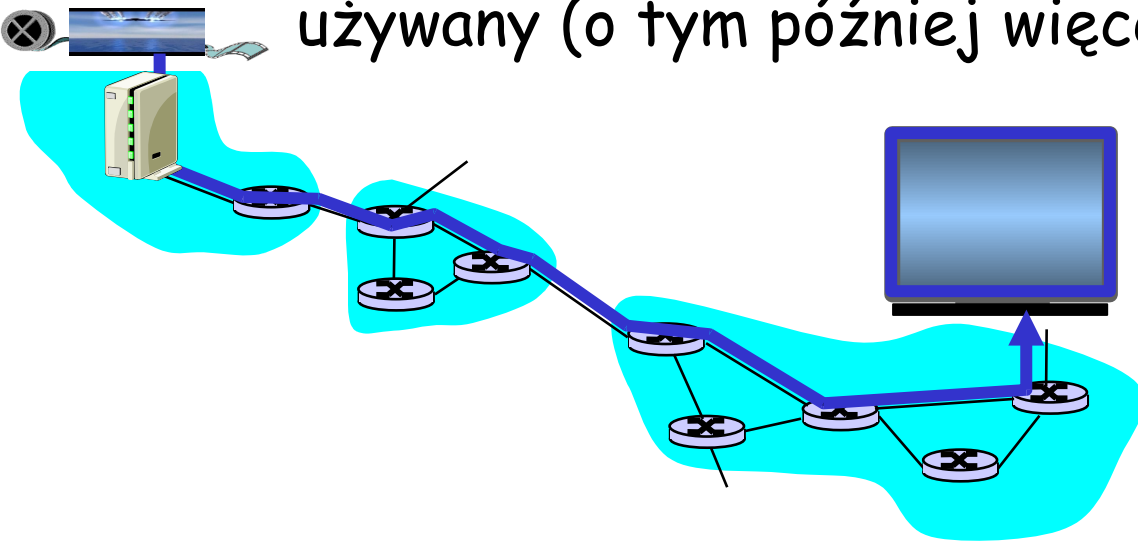


przesyłanie strumieniowe: w tym czasie klient odtwarza pierwszą część obrazu, podczas gdy serwer wciąż przesyła dalszą część obrazu

Przesyłanie strumieniowe przechowywanych multimediiów: Interaktywność



- *Funkcjonalność odtwarzacza wideo*: klient może pauzować, przewijać wstecz, szybko do przodu, przycisnąć pasek suwaka
 - 10 sekund opóźnienia początkowego OK
 - 1-2 sekundy do poskutkowania komendy OK
 - RTSP (*Real Time Streaming Protocol*) często używany (o tym później więcej)



- *Ograniczenie czasowe dla danych pozostałych do przestania*: w czasie wystarczającym na odtworzenie

Przesyłanie strumieniowe multimedialnych na żywo



Przykłady:

- Internetowy talk show radiowy
- Impreza sportowa na żywo

Przesyłanie strumieniowe

- Bufor odtwarzania
- Odtwarzanie może opóźnić się o dziesiętne sekundy po transmisji
- Wciąż ma ograniczenie czasowe

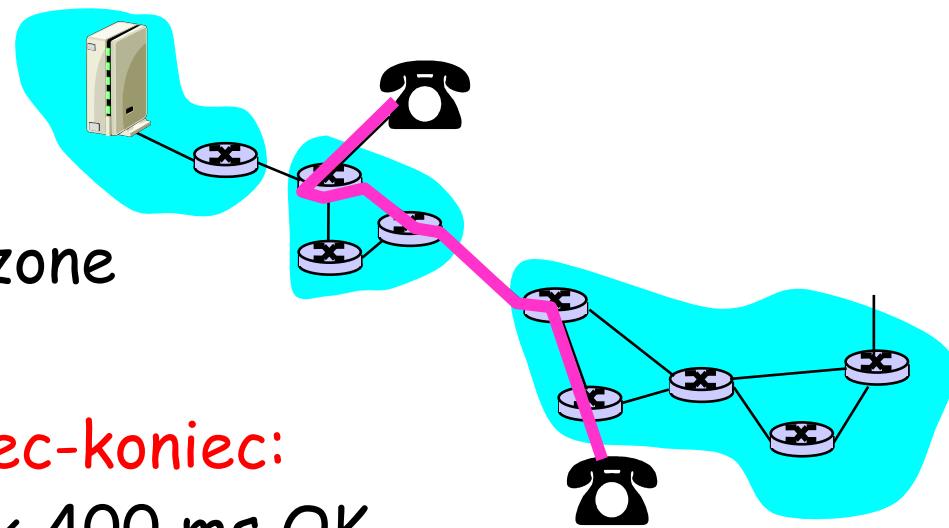
Interaktywność

- Brak możliwości szybkiego przewijania do przodu
- Możliwość przewijania wstecz, pauzy!

Interaktywne multimedia czasu rzeczywistego



- **aplikacje:** telefonia IP, wideokonferencje, rozproszone światy interaktywne
- **Wymagania opóźnienia koniec-koniec:**
 - audio: < 150 ms dobrze, < 400 ms OK
 - Zawiera warstwy wyższe (enkapsulacja) i opóźnienia sieciowe
 - Większe opóźnienia zauważalne, zmniejszają interaktywność
- **Inicjowanie sesji**
 - W jaki sposób dzwoniący ogłasza swój adres IP, numer portu, algorytmy kodowania?



Multimedia w dzisiejszym Internecie



TCP/UDP/IP: "usługi best-effort"

- **brak** gwarancji co do opóźnienia, utraty danych



? ? ? ? ?
Ale podobno aplikacje multimedialne wymagają,
żeby jakość usług i poziom wydajności były
? wysokie! ? ?



Dzisiejsze internetowe aplikacje multimedialne
używają technik z warstwy aplikacji do
ograniczania (w miarę możliwości) skutków
opóźnienia, utraty danych

Jak powinien ewoluować Internet, żeby lepiej wspierać multimedia?



Filozofia zintegrowanych usług:

- ❑ Fundamentalne zmiany w Internecie, żeby aplikacje mogły rezerwować szerokość pasma na całej drodze transmisji
- ❑ Wymaga nowego, złożonego oprogramowania hostów & ruterów

Laissez-faire

- ❑ Brak większych zmian
- ❑ Większa szerokość pasma w razie potrzeby
- ❑ Dystrybucja zawartości, sieci nakładkowe
 - Warstwa aplikacji

Filozofia zróżnicowanych usług:

- ❑ Mniej zmian w infrastrukturze internetowej, ale zapewniają usługę 1 i 2 klasy.



Jakie jest twoje zdanie?

Kilka słów o kompresji dźwięku



- Sygnał analogowy próbkowany ze stałą prędkością
 - telefon: 8,000 próbki/sekundę
 - Muzyka CD: 44,100 próbki/sekundę
 - Każda próbka skwantowana, tj., zaokrąglona
 - np., $2^8=256$ możliwe skwantowane wartości
 - Każda skwantowana wartość reprezentowana przez bity
 - 8 bitów na 256 wartości
 - Przykład: 8,000 próbek/sekundę, 256 skwantowanych wartości--> 64,000 b/s
 - Pulse Code Modulation (PCM)
 - odbiornik konwertuje z powrotem na sygnał analogowy:
 - Pewne obniżenie jakości
- ### Przykładowe prędkości
- CD: 1.411 Mb/s
 - MP3: 96, 128, 160 kb/s
 - GSM: 13 kb/s
 - Telefonia internetowa: 5.3 - 13 kb/s

Kilka słów o kompresji obrazu



- ❑ Video jest sekwencją obrazów wyświetlanych ze stałą prędkością
 - np. 24 obrazy/sekundę
- ❑ Obraz cyfrowy jest układem pikseli
- ❑ Każdy piksel jest reprezentowany przez bity
- ❑ Redundancja
 - przestrzenna
 - czasowa

Przykłady:

- ❑ MPEG 1 (CD-ROM) 1.5 Mb/s
- ❑ MPEG2 (DVD) 3-6 Mb/s
- ❑ MPEG4 (często używane w Internecie, < 1 Mb/s)

W trakcie badań:

- ❑ Video warstwowe (skalowalne)
 - adaptacja warstw do dostępnej szerokości pasma

Zarys wykładu o komunikacji multimedialnej w sieciach IP

- Aplikacje sieci z jakością usług
- Przesyłanie strumieniowe przechowywanych plików audio i wideo
 - RTSP
- Multimedia czasu rzeczywistego: studium przypadku telefonii internetowej
- Protokoły dla interaktywnych aplikacji czasu rzeczywistego
 - RTP, RTCP
 - SIP
- Poza best-effort
- Mechanizmy szeregowania i kontroli

Przesyłanie strumieniowe przechowywanych multimediiów



Techniki przesyłania

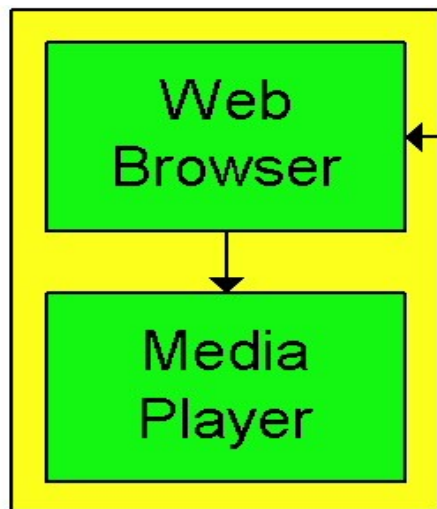
strumieniowego na poziomie aplikacji do jak najlepszego wykorzystania usług best-effort:

- Buforowanie po stronie klienta
- Stosowanie UDP albo TCP
- Wielokrotne kodowanie multimediiów

Odtwarzacz

- Usuwanie rozsynchronizowania
- dekompresja
- Ukrywanie błędów
- graficzny interfejs użytkownika dla sterowania interaktywnością

Internetowe multimedia: najprostsze podejście



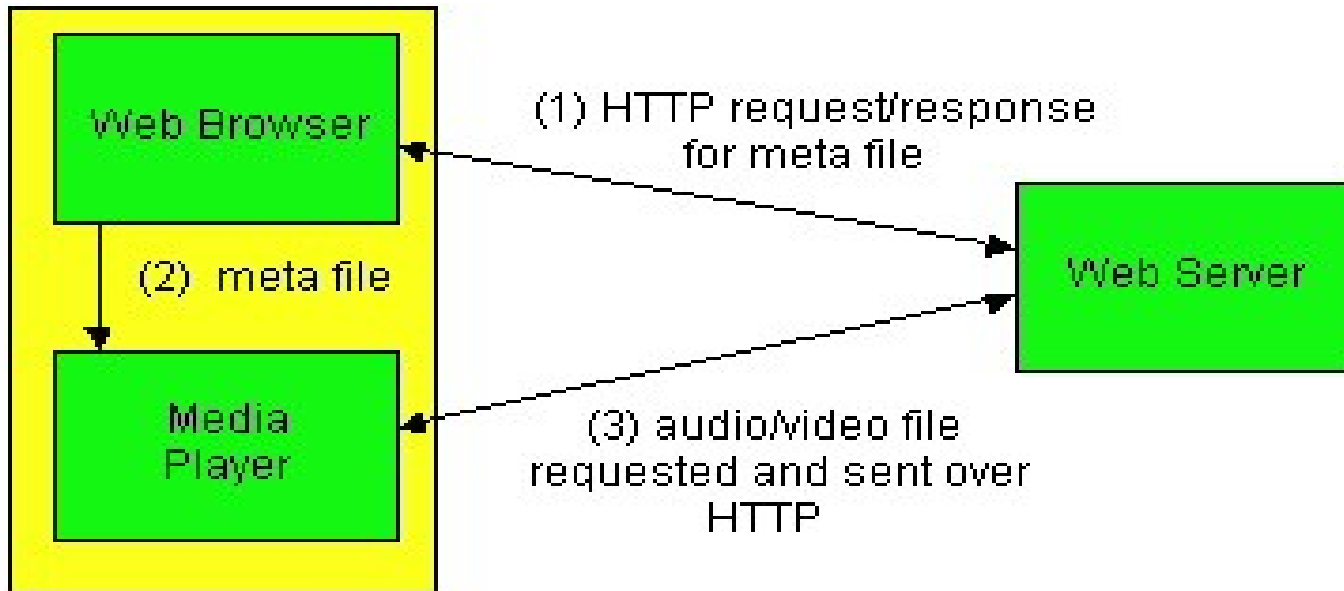
client

- ❑ dźwięk lub obraz przechowywane w pliku
- ❑ pliki przenoszone jako obiekt HTTP przez TCP
 - otrzymywane w całości u klienta
 - następnie przekazywane do odtwarzacza

dźwięk, obraz nie przetwarzane strumieniowo:

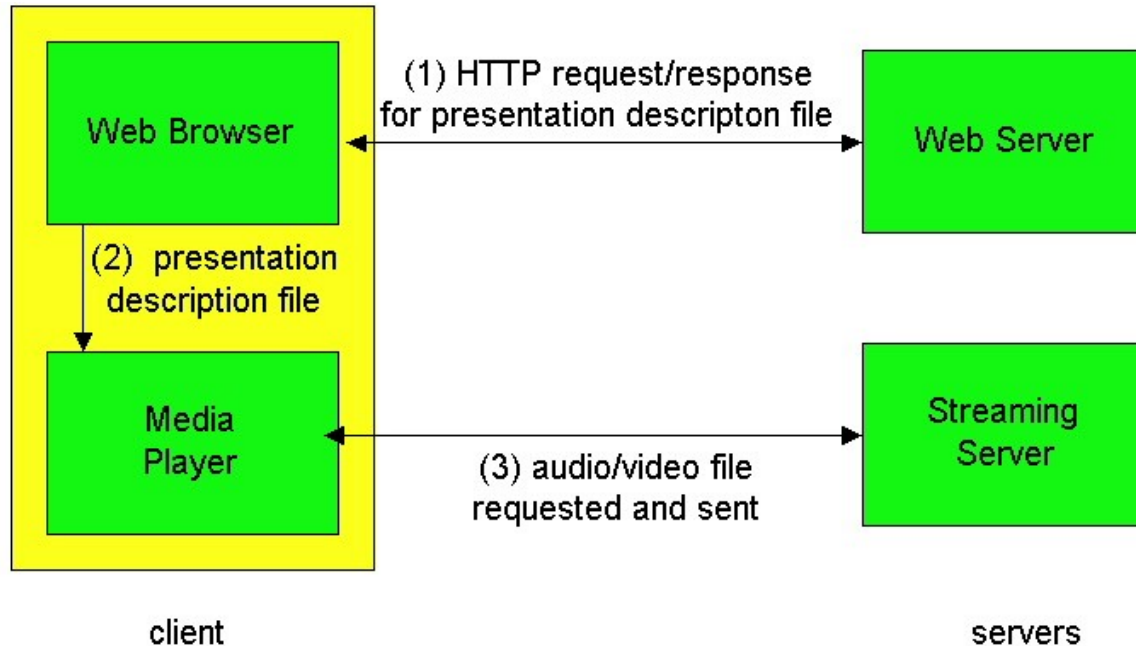
- ❑ brak "strumienia", długie opóźnienia do odtwarzania!

Internetowe multimedia: podejście oparte na przesyłaniu strumieniowym



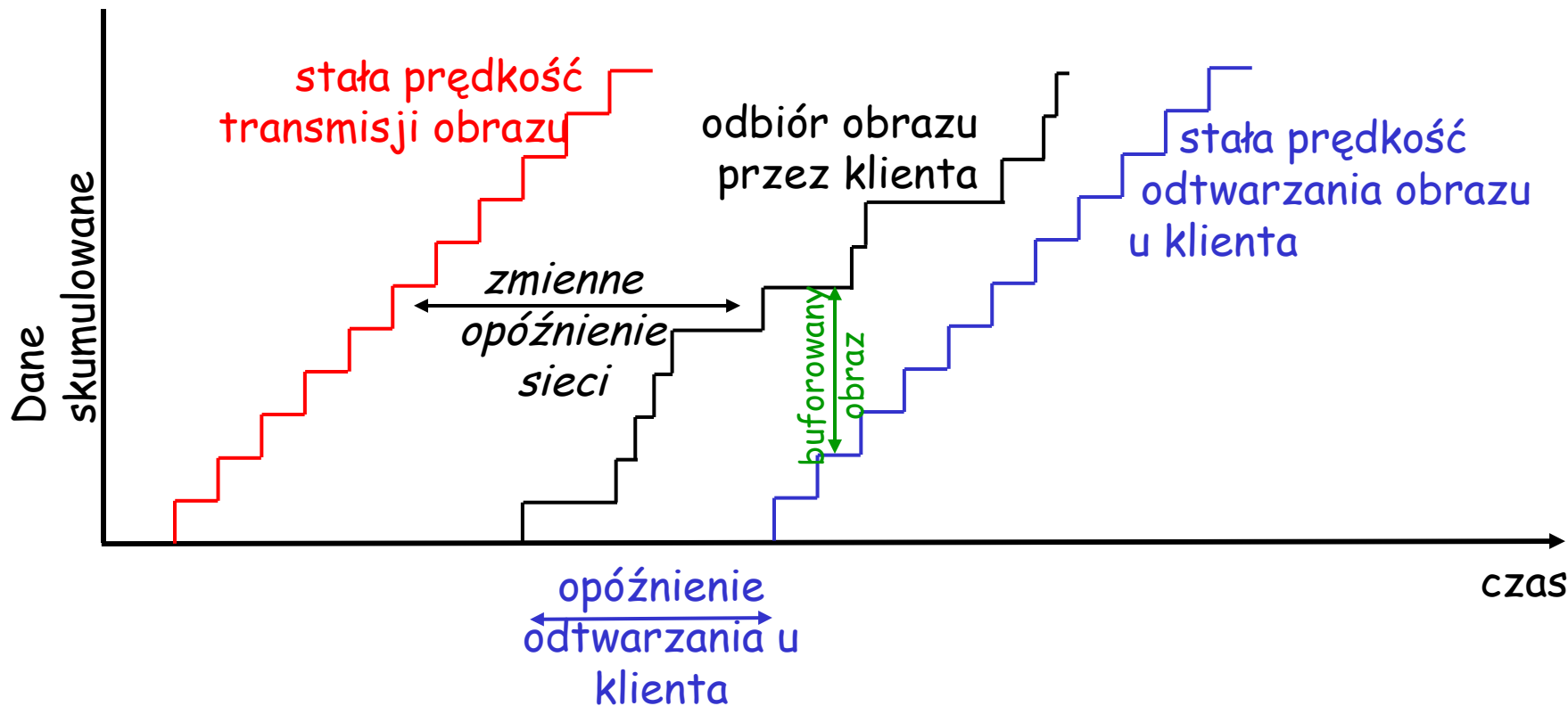
- ❑ przeglądarka otrzymuje **metaplik**
- ❑ przeglądarka uruchamia odtwarzacz, przekazując metaplik
- ❑ odtwarzacz kontaktuje się z serwerem
- ❑ serwer **strumieniuje** dźwięk/obraz do odtwarzacza

Przesyłanie strumieniowe z serwera przesyłania strumieniowego



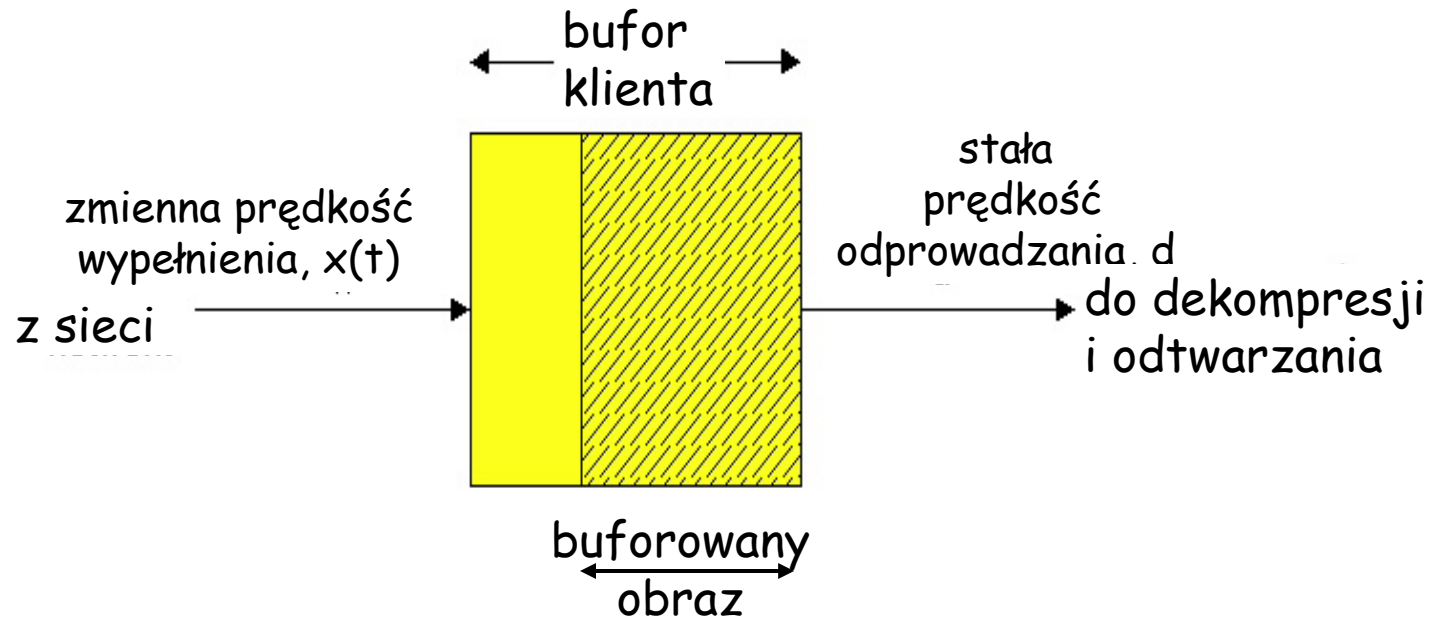
- Ta architektura dopuszcza protokół inny niż HTTP między serwerem a odtwarzaczem
- Może również używać UDP zamiast TCP.

Przesyłanie strumieniowe multimedialnych: Buforowanie u klienta



- Buforowanie po stronie klienta, opóźnienie odtwarzania kompensuje dodane opóźnienie sieci, zmienność opóźnienia

Przesyłanie strumieniowe multimediiów: Buforowanie u klienta



- Buforowanie po stronie klienta, opóźnienie odtwarzania kompensuje opóźnienie dodane sieci, zmienność opóźnienia

Przesyłanie strumieniowe multimedialnych: UDP czy TCP?



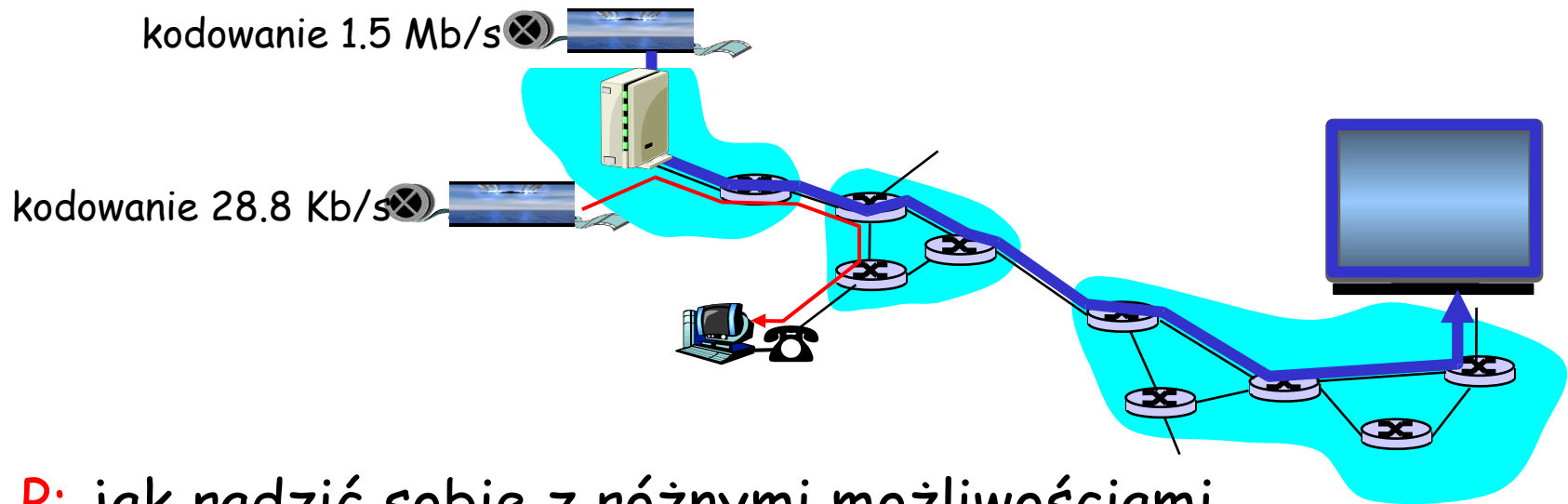
UDP

- ❑ serwer wysyła z prędkością odpowiednią dla klienta (niezależnie od przeciążenia sieci!)
 - Często prędkość wysyłki = prędkość kodowania = prędkość stała
 - następnie, prędkość wypełnienia = prędkość stała - utrata pakietu
- ❑ Krótkie opóźnienie odtwarzania (2-5 sekund) kompensujące zmienność opóźnienia sieciowego
- ❑ Naprawa błędów: na ile pozwala na to czas

TCP

- ❑ Wysyłanie z maksymalną możliwą prędkością w TCP
- ❑ Prędkość wypełnienia waha się ze względu na kontrolę przeciążenia TCP
- ❑ Większe opóźnienie odtwarzania: gładkie tempo dostarczania TCP
- ❑ HTTP/TCP łatwiej przechodzi przez zapory ogniowe

Przesyłanie strumieniowe multimedialnych : prędkość (prędkości) klienta



P: jak radzić sobie z różnymi możliwościami klienta w zakresie prędkości odbioru?

- łącze komutowane 28.8 Kb/s
- Ethernet 100Mb/s

O: serwer przechowuje, transmituje wiele kopii obrazu kodowanych z różnymi prędkościami

Sterowanie przesyłaniem strumieniowym mediów przez klienta: RTSP



HTTP

- Nie zajmuje się treścią multimedialną
- Brak komend do szybkiego przewijania w przód, itp.

RTSP: RFC 2326

- Protokół warstwy aplikacji klient-serwer.
- Do sterowania wyświetlaniem przez użytkownika: przewijanie wstecz, szybko do przodu, pauza, wznowienie, zmiana pozycjonowania, itp....

Czego nie robi:

- Nie określa, jak dźwięk/obraz jest kodowany, kompresowany i enkapsułowany (może być w RTP lub innym protokole) do przesyłania strumieniowego w sieci
- Nie ogranicza sposobu przesyłania strumieniowanych mediów: mogą być przesyłane przez UDP lub TCP
- Nie określa, jak odtwarzacz buforuje dźwięk/obraz

RTSP: sterowanie poza pasmem



FTP używa kanału sygnalizacyjnego "poza pasmem":

- ❑ Plik jest transmitowany w ciągu jednego połączenia TCP.
- ❑ Informacje sygnalizacyjne (zmiany katalogu, usunięcie pliku, zmiana nazwy pliku, itp.) przesyła się w ramach oddzielnego połączenia TCP.
- ❑ Kanały "poza pasmem" i "w obrębie pasma" używają różnych numerów portów.

Wiadomości RTSP przesyła się również poza pasmem:

- ❑ Wiadomości sygnalizacyjne RTSP używają numerów portów innych niż strumień mediów: poza pasmem.
 - Port 554
- ❑ Strumień mediów uważa się za pozostający "w obrębie pasma".



Przykład RTSP

Scenariusz:

- ❑ Metaplik przekazywany do przeglądarki internetowej
- ❑ Przeglądarka uruchamia odtwarzacz
- ❑ Odtwarzacz nawiązuje połączenie sygnalizacyjne RTSP, połączenie danych do serwera przesyłania strumieniowego

Przykład metapliku

```
<title>Twister</title>
```

```
<session>
```

```
  <group language=en lipsync>
```

```
    <switch>
```

```
      <track type=audio
```

```
        e="PCMU/8000/1"
```

```
        src = "rtsp://audio.example.com/twister/audio.en/lofi">
```

```
      <track type=audio
```

```
        e="DVI4/16000/2" pt="90 DVI4/8000/1"
```

```
        src="rtsp://audio.example.com/twister/audio.en/hifi">
```

```
    </switch>
```

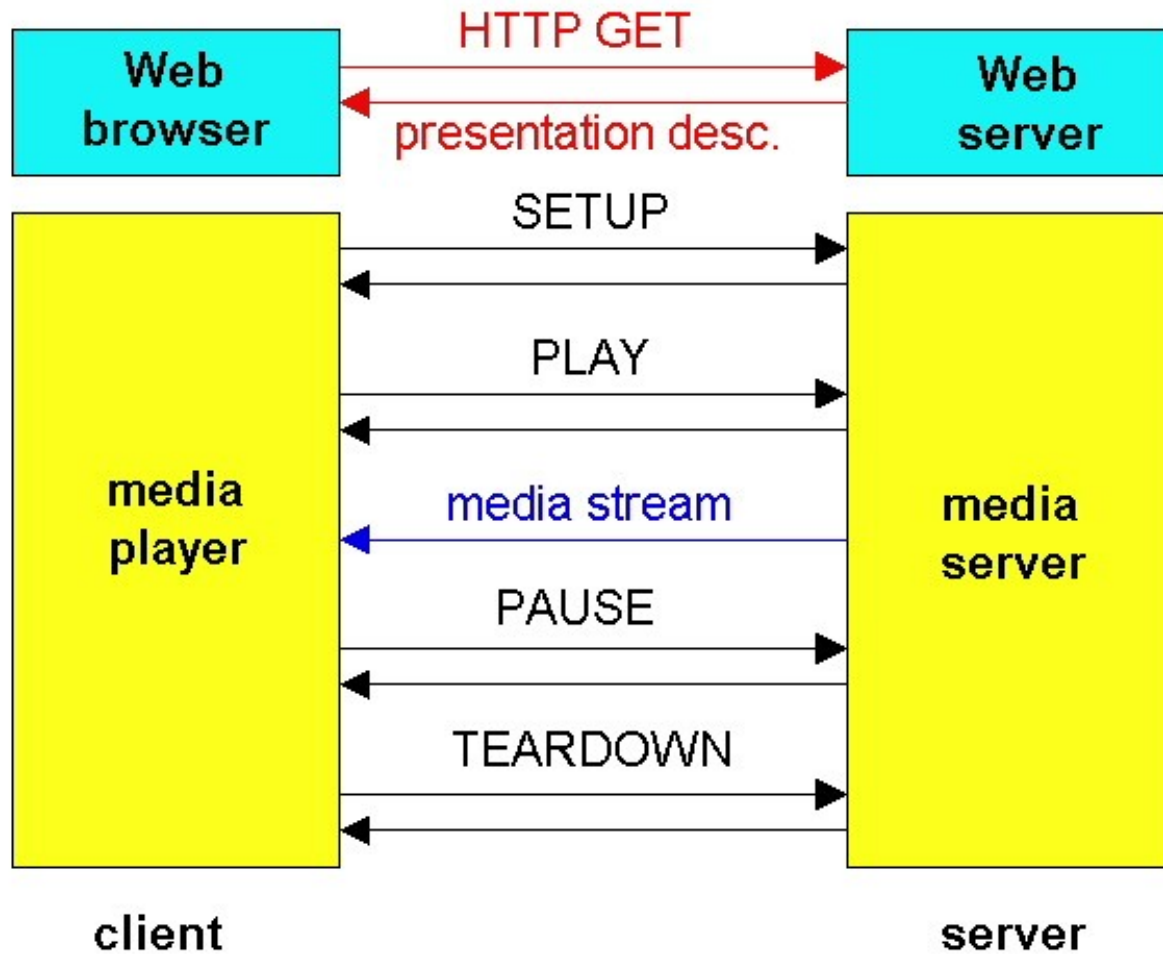
```
  <track type="video/jpeg"
```

```
    src="rtsp://video.example.com/twister/video">
```

```
  </group>
```

```
</session>
```

Działanie RTSP



Przykład wymiany RTSP



C: SETUP rtsp://audio.example.com/twister/audio RTSP/1.0
Transport: rtp/udp; compression; port=3056; mode=PLAY

S: RTSP/1.0 200 1 OK
Session 4231

C: PLAY rtsp://audio.example.com/twister/audio.en/lofi RTSP/1.0
Session: 4231
Range: npt=0-

C: PAUSE rtsp://audio.example.com/twister/audio.en/lofi RTSP/1.0
Session: 4231
Range: npt=37

C: TEARDOWN rtsp://audio.example.com/twister/audio.en/lofi RTSP/1.0
Session: 4231

S: 200 3 OK

Zarys wykładu o komunikacji multimedialnej w sieciach IP

- Aplikacje sieci z jakością usług
- Przesyłanie strumieniowe przechowywanych plików audio i wideo
 - RTSP
- **Multimedia czasu rzeczywistego: studium przypadku telefonii internetowej**
- Protokoły dla interaktywnych aplikacji czasu rzeczywistego
 - RTP, RTCP
 - SIP
- Poza best-effort
- Mechanizmy szeregowania i kontroli

Interaktywne aplikacje czasu rzeczywistego

- Telefon PC-2-PC
 - Zapewniają to usługi komunikatora wiadomości
- PC-2-telefon
 - Klawiatura telefoniczna
 - Net2phone
- Wideokonferencja z kamerami internetowymi

Teraz przyjrzymy się
szczegółowo
przykładowi telefonu
internetowego PC-2-PC

Interaktywne multimedia: telefon internetowy



Wprowadzenie telefonu internetowego jako przykładu

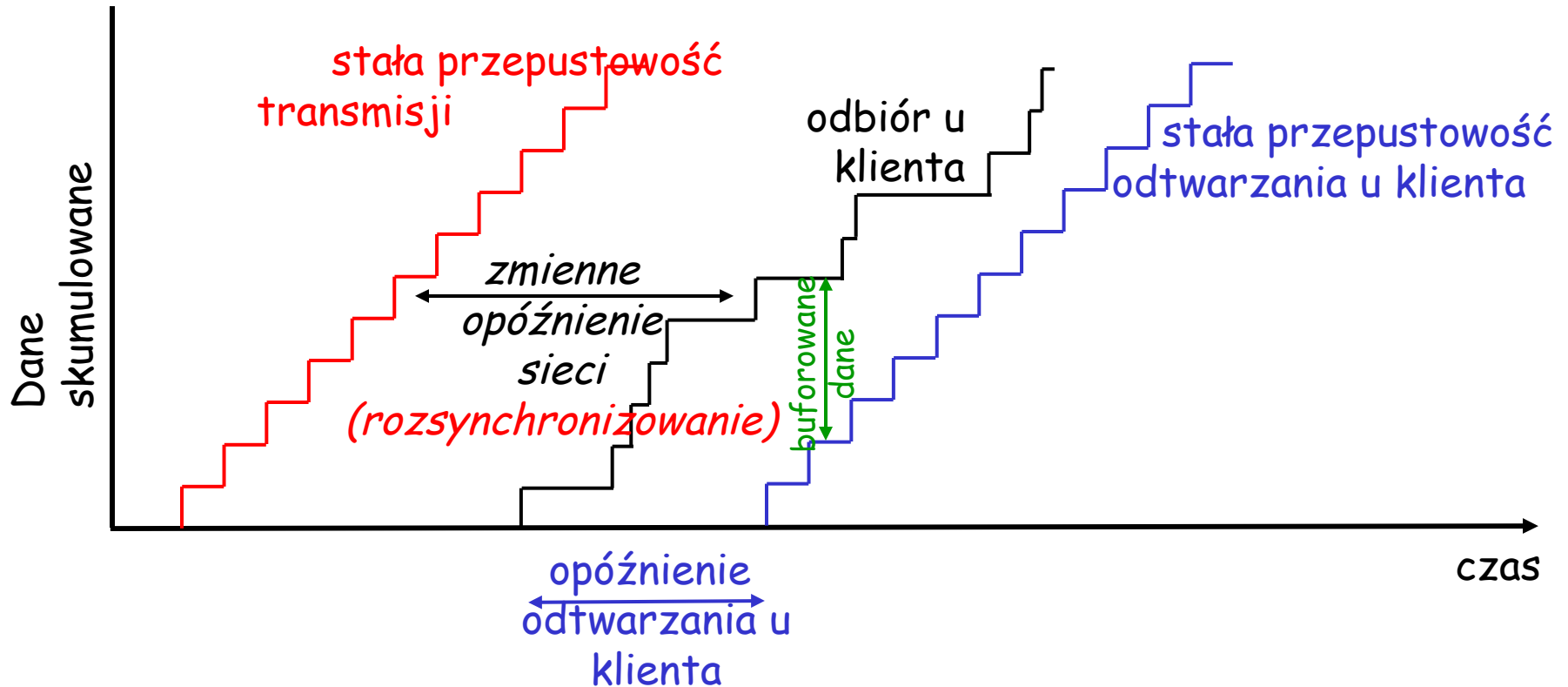
- ❑ Dźwięk mówiącego: naprzemiennie porcje mowy, okresy ciszy.
 - 64 kb/s w trakcie porcji mowy
- ❑ pakiety generowane tylko w trakcie strumieni mowy
 - Fragmenty 20 ms z prędkością 8 Kb/s: dane 160 bajtów
- ❑ Do każdego fragmentu dodawany jest nagłówek z warstwy aplikacji.
- ❑ Fragment+nagłówek wbudowany w segment UDP.
- ❑ aplikacja wysyła segment UDP do każdego gniazda co 20 ms w trakcie porcji mowy.

Telefon internetowy: strata pakietu i opóźnienie



- ❑ **Strata w sieci:** utracony pakiet IP z powodu przeciążenia sieci (przepelnienie bufora rutera)
- ❑ **Strata z powodu opóźnienia:** pakiet IP dociera za późno, żeby możliwe było odtworzenie w odbiorniku
 - opóźnienia: przetwarzanie, kolejkowanie w sieci; opóźnienia systemu końcowego (nadawca, odbiorca)
 - Typowe maksymalne tolerowane opóźnienie: 400 ms
- ❑ **Tolerancja utraty:** zależnie od kodowania głosu, maskowania strat, odsetek strat pakietu pomiędzy 1% a 10% może być tolerowany.

Zmienność opóźnienia



- rozważmy opóźnienia od końca do końca dwóch kolejnych pakietów: różnica może być większa lub mniejsza od 20 ms

Telefon internetowy: stałe opóźnienie odtwarzania



- Odbiorca stara się odtworzyć każdy fragment w dokładnym czasie q ms po wygenerowaniu tego fragmentu.
 - Fragment ma znacznik czasowy t : odegrać fragment w czasie $t+q$.
 - Fragment zostaje odebrany po $t+q$: dane zostają odebrane za późno, żeby je odtworzyć, dane "utracone"
- Sterowanie za pomocą q :
 - Duże q : mniej strat pakietów
 - Małe q : lepsze doświadczenie interaktywne

Stałe opóźnienie odtwarzania

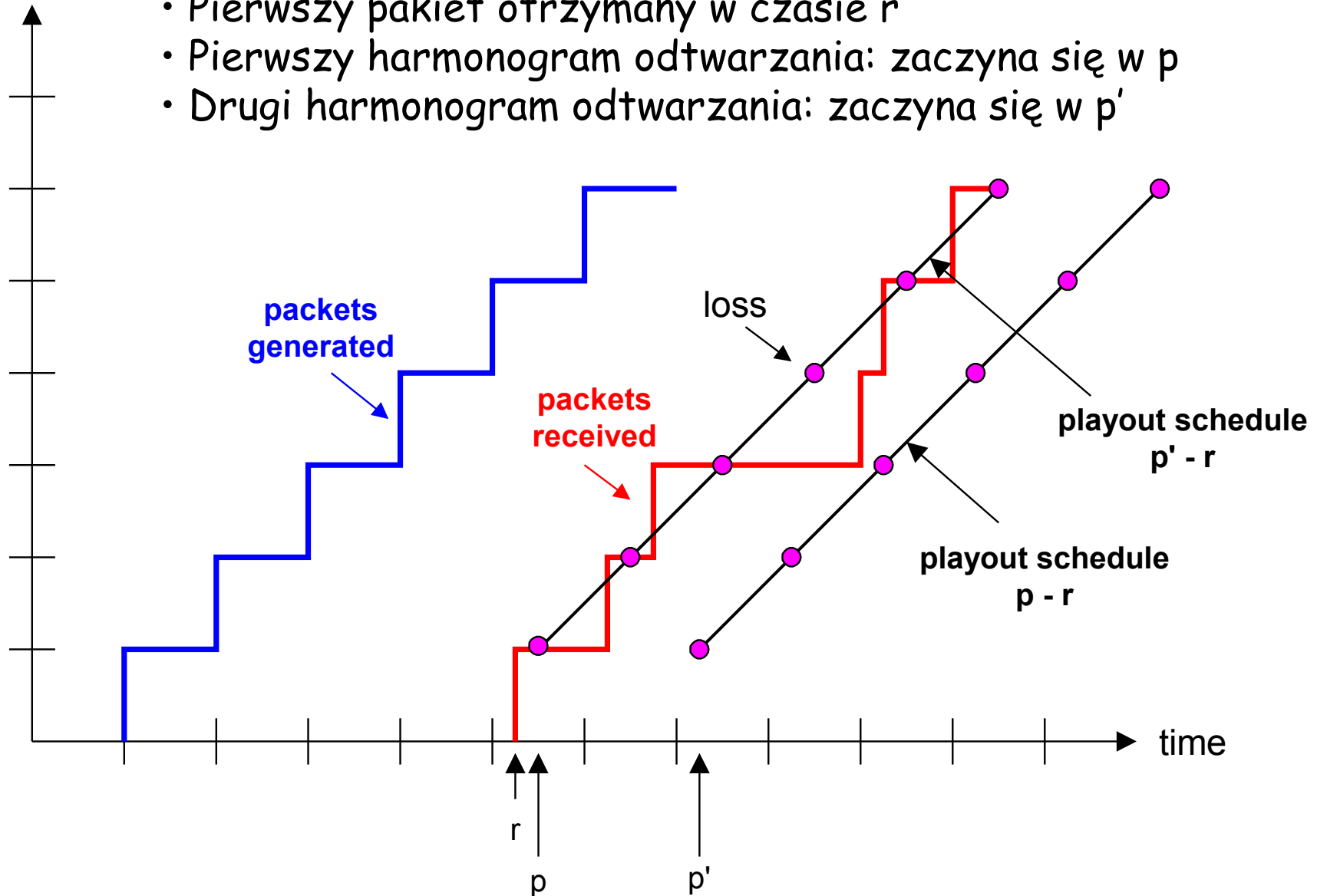


packets • Nadawca generuje pakiety co 20 ms w trakcie porcji mowy.

• Pierwszy pakiet otrzymany w czasie r

• Pierwszy harmonogram odtwarzania: zaczyna się w p

• Drugi harmonogram odtwarzania: zaczyna się w p'



Adaptacyjne opóźnienie odtwarzania, I



- **Cel:** minimalizacja opóźnienia odtwarzania, utrzymywanie niskiego odsetka strat wynikających z opóźnienia
- **Podejście:** adaptacyjna korekta opóźnienia odtwarzania:
 - Oszacować opóźnienie sieci, skorygować opóźnienie odtwarzania na początku każdej porcji mowy.
 - Okresy ciszy kompresowane i wydłużane.
 - Fragmenty wciąż odtwarzane co 20 ms w trakcie porcji mowy.

t_i = znacznik czasowy pakietu i

r_i = czas, kiedy pakiet i jest odbierany przez odbiorcę

p_i = czas, kiedy pakiet i jest odtwarzany u odbiorcy

$r_i - t_i$ = opóźnienie w sieci pakietu i

d_i = estymacja po otrzymaniu pakietu i średniego opóźnienia

Dynamiczna estymacja średniego opóźnienia u odbiorcy:

$$d_i = (1 - u)d_{i-1} + u(r_i - t_i)$$

gdzie u jest ustaloną stałą (np., $u = .01$).

Adaptacyjne opóźnienie odtwarzania II



Przydatne również do obliczania średniego odchylenia opóźnienia, v_i :

$$v_i = (1 - u)v_{i-1} + u |r_i - t_i - d_i|$$

Szacunki d_i i v_i oblicza się dla każdego otrzymanego pakietu, chociaż używa się ich tylko na początku porcji mowy.

Dla pierwszego pakietu w porcji mowy czas odtwarzania jest następujący:

$$p_i = t_i + d_i + Kv_i$$

gdzie K jest dodatnią stałą.

Pozostałe pakiety w porcji mowy są odtwarzane okresowo

Adaptacyjne odtwarzanie, III



- P:** W jaki sposób odbiorca ustala, czy pakiet jest pierwszy w porcji mowy?
- Jeżeli nie ma utraty, odbiorca patrzy na kolejne znaczniki czasowe.
 - Różnica kolejnych znaczników > 20 ms --> zaczyna się porcja mowy.
 - W razie możliwości utraty danych odbiorca musi patrzeć zarówno na znaczniki czasowe, jak i na kolejne numery.
 - Różnica kolejnych znaczników > 20 ms **oraz** kolejne numery bez luk --> zaczyna się porcja mowy.

Odtwarzanie po utracie pakietu (1)



Kodowanie nadmiarowe (FEC): prosty schemat

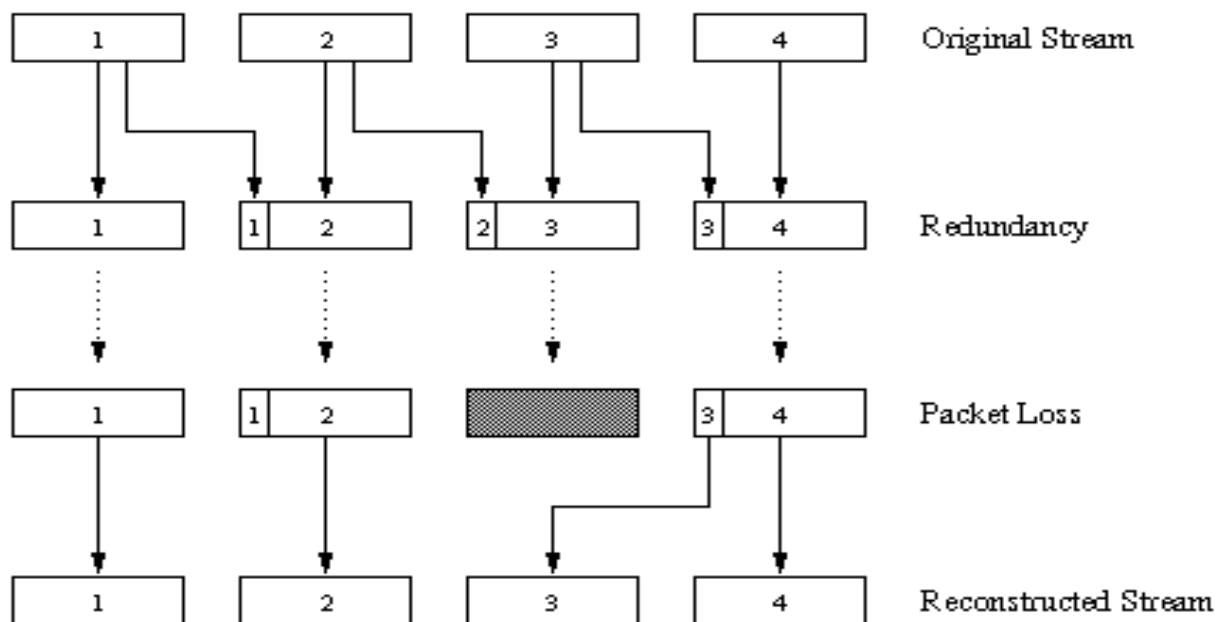
- Dla każdej grupy n części utworzyć część nadmiarową przez operację XOR dla n poprzednich części
- Wysłać $n+1$ części, zwiększając zużytą przepustowość o współczynnik $1/n$.
- można zrekonstruować pierwotne n części, jeżeli spośród $n+1$ części została utracona najwyżej jedna część
- Opóźnienie odtwarzania musi zostać dopasowane do czasu, aby otrzymać wszystkie $n+1$ pakietów
- Sterowanie:
 - Zwiększyć n , mniej zmarnowanej przepustowości
 - Zwiększyć n , dłuższe opóźnienie odtwarzania
 - Zwiększyć n , wyższe prawdopodobieństwo, że 2 lub więcej części zostanie utraconych

Odtwarzanie po utracie pakietu (2)



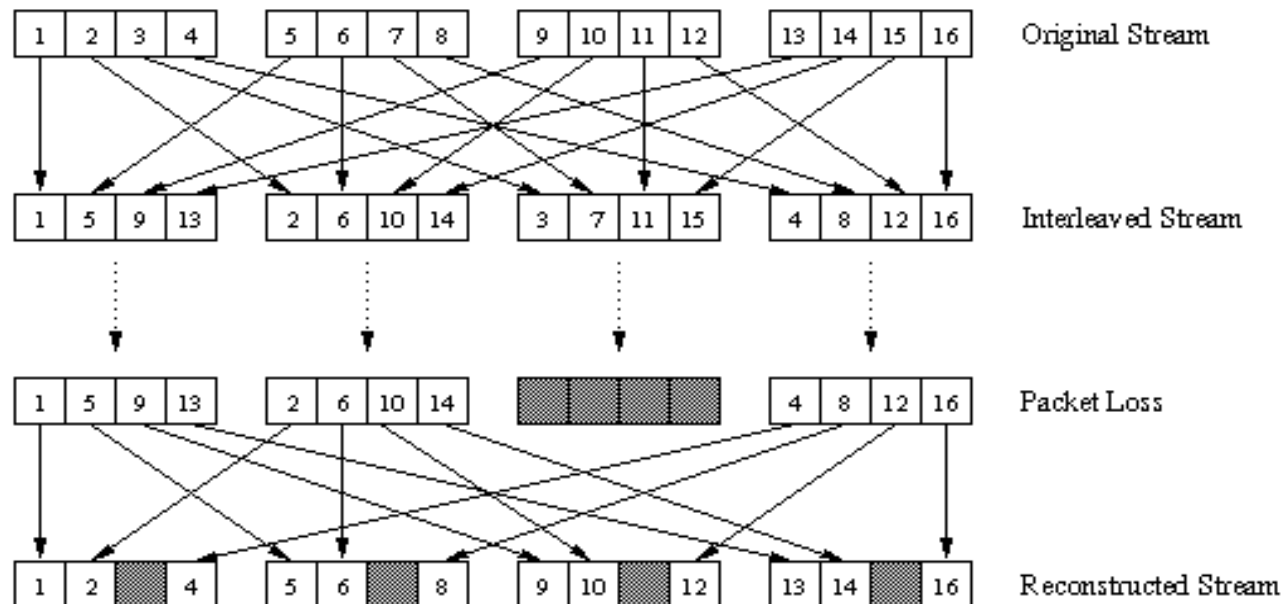
Drugi schemat FEC

- "nakładany strumień niższej jakości"
- wysłać strumień obrazu o niższej rozdzielczości jako redundantne informacje
- na przykład znamionowy strumień PCM z prędkością 64 kb/s i redundantny strumień GSM z prędkością 13 kb/s.



- Gdziekolwiek występuje strata niesekwencyjna, odbiorca może zamaskować stratę.
- może również dodać fragment (n-1) i (n-2) o małej przepustowości

Odtwarzanie po utracie pakietu (3)



Przeplatanie

- części są rozbijane na mniejsze jednostki
- na przykład 4 jednostki po 5 ms
- Pakiet zawiera małe jednostki z różnych części
- jeżeli pakiet zostaje utracony, wciąż pozostaje większość każdej części
- Nie ma obciążenia redundancją
- Ale zwiększa się opóźnienie odtwarzania

Podsumowanie: Internetowe multimedia: kompletne wyposażenie



- ❑ **Użyć UDP**, aby uniknąć kontroli przeciążenia TCP (opóźnienia) dla ruchu podatnego na czas
- ❑ Adaptacyjne opóźnienie odtwarzania po stronie klienta: aby skompensować opóźnienie
- ❑ Strona serwera dopasowuje przepustowość **strumienia** do przepustowości dostępnej ścieżki od klienta do serwera
 - Wybór spośród wstępnie kodowanej prędkości strumienia
 - Dynamiczne tempo kodowania serwera
- ❑ Usuwanie skutków strat (korzystając z UDP)
 - FEC, przeplatanie
 - retransmisje, o ile pozwala czas
 - Maskowanie błędów: powtórzenie pobliskich danych, interpolacja