

# Wprowadzenie do telefonii VoIP

Teoria vs real life

*M. Żyromski, A. Ciarkowski*  
*Wirtualna Polska S.A.*

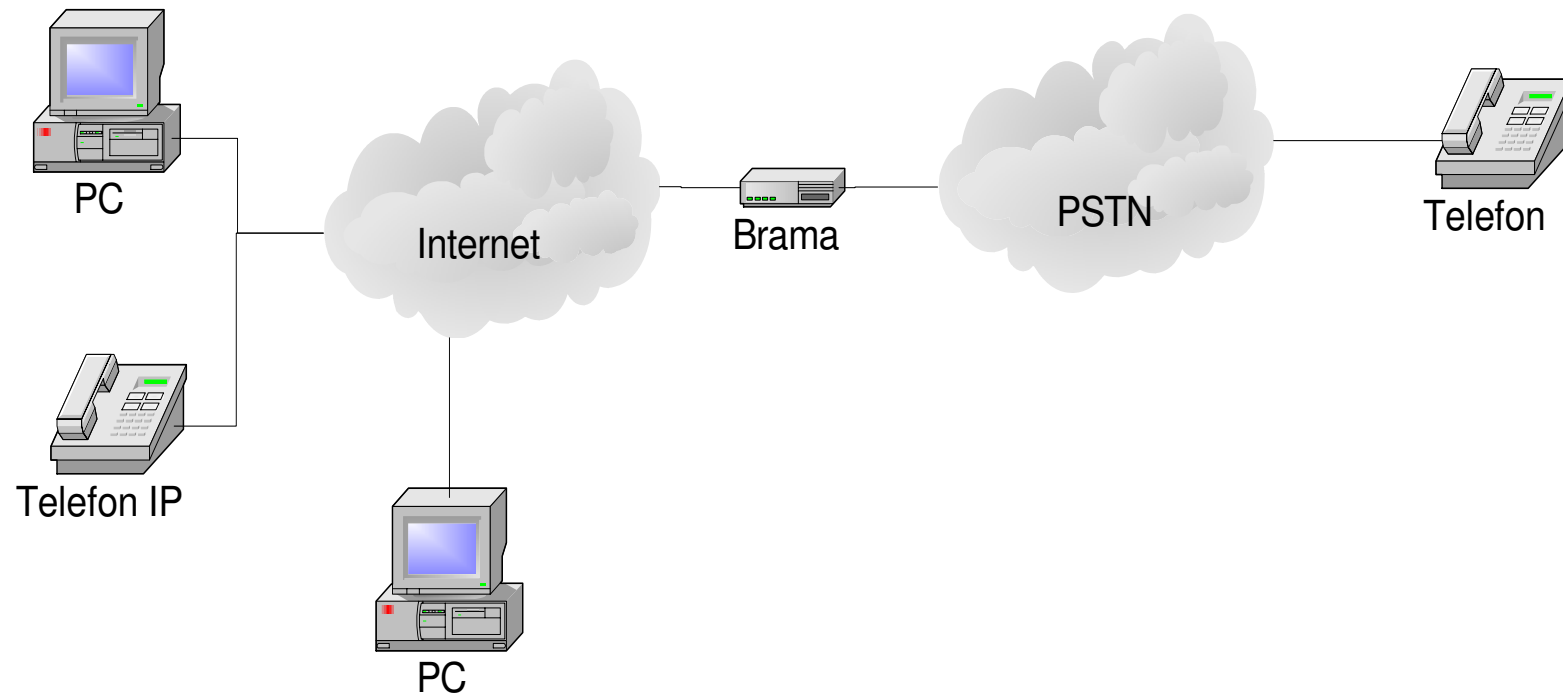
# Plan prezentacji

- Wprowadzenie
- Zagadnienia sieciowe
- Protokoły
- Transmisja mediów w telefonii VoIP

## VoIP – z czym to się je?

- VoIP – Voice over **IP**
- Grupy zastosowań
  - pc2phone, phone2pc
  - pc2pc
  - phone2phone

# Schemat ogólny



---

# VoIP – wyzwania i problemy

- Opóźnienia
- Jitter
- NATy, firewalle
- Echo
- Straty danych
  
- Bezpieczeństwo

---

# Zagadnienia sieciowe - plan

- Wybór protokołu transportowego (UDP vs TCP)
- NATy – problemy, sposoby przechodzenia

---

# UDP vs TCP

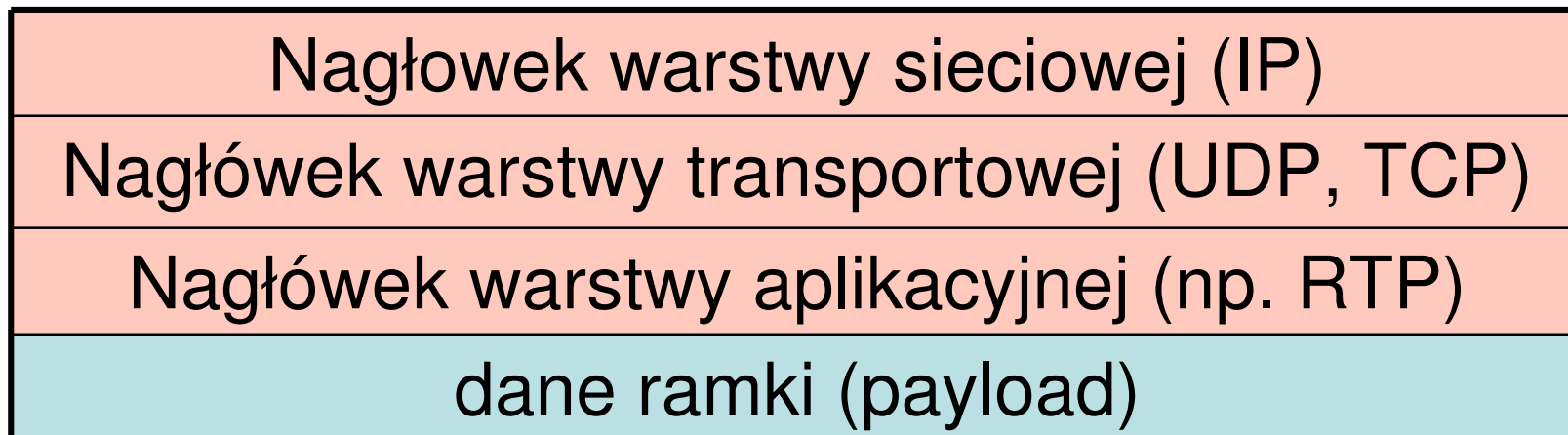
## UDP

- Nagłówek 8B
- Bezpołączeniowy
- Brak mechanizmu retransmisji i kontroli przepływu
  - Mniejsze opóźnienia
  - Straty
  - Zaburzenia sekwencyjności

## TCP

- Nagłówek 20B
  - Połączeniowy
  - Mechanizm retransmisji i kontroli przepływu
    - Większe Opóźnienia
    - Brak strat
    - Zachowanie sekwencyjności
- 
- Możliwość „zagłodzenia” ruchu TCP przez UDP
  - Konfiguracja sieci może uniemożliwiać ruch UDP

# Narzuty protokołu

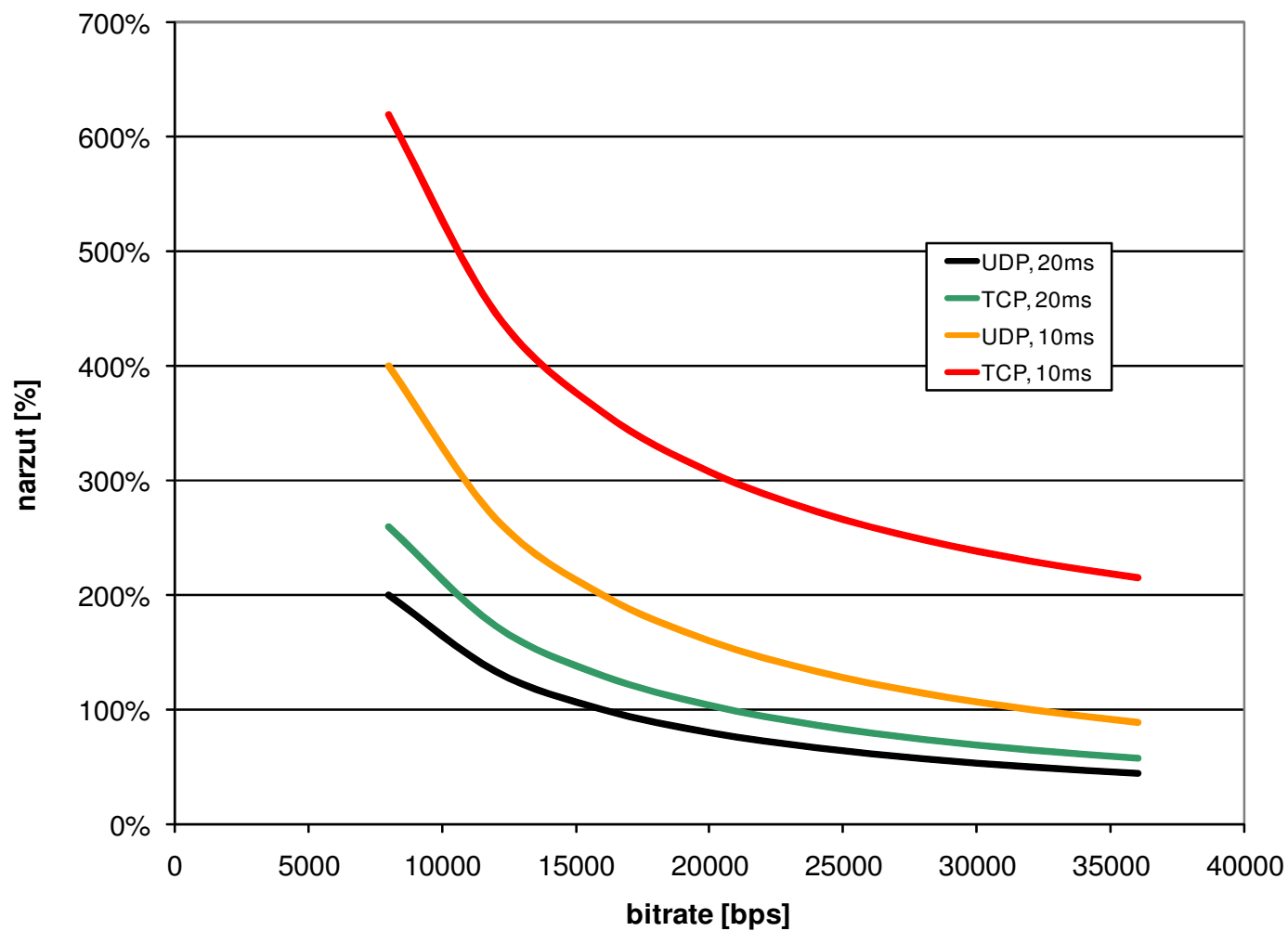


Przykład: bitrate 16kbps, ramka 20ms

Ramka	= 40B
+ nagłówek RTP	= 52B
+ nagłówek UDP	= 60B
+ nagłówek IP	= <b>80B</b>

**Rzeczywisty bitrate: 32kbps**

# Narzuty protokołu

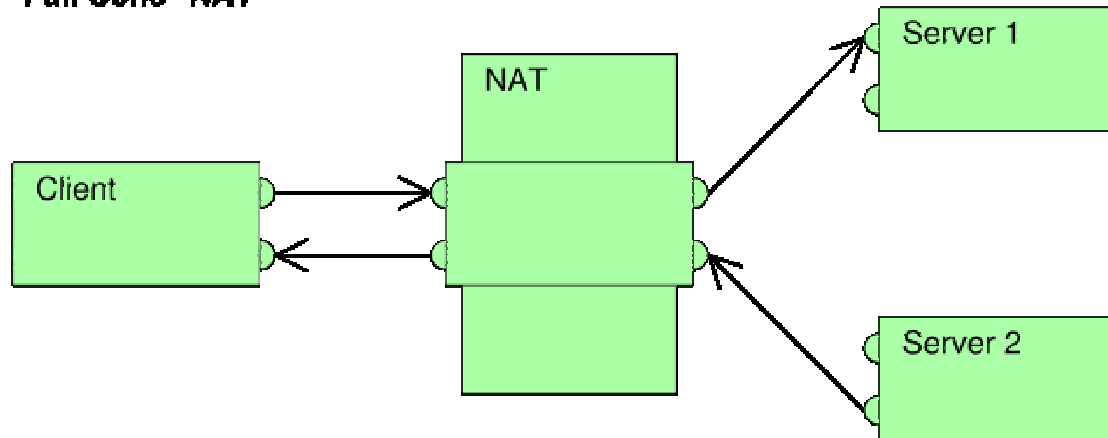


# NAT

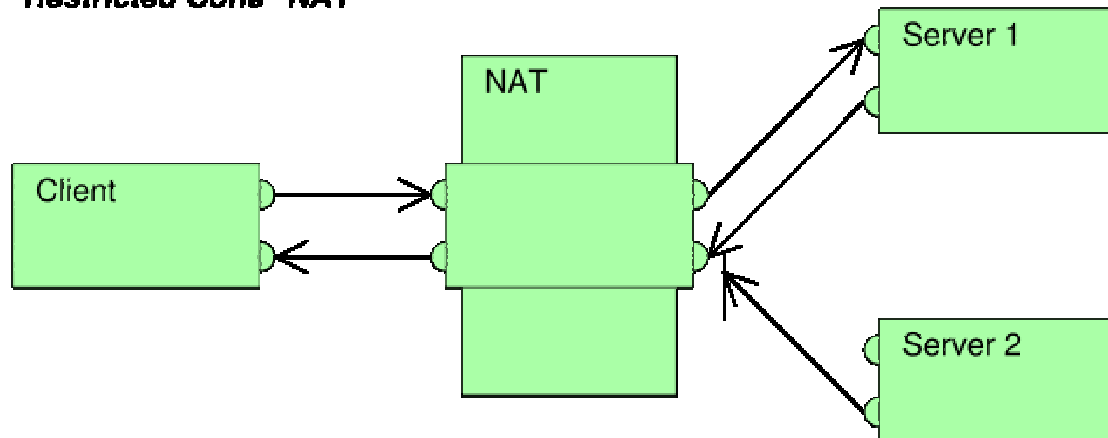
- Ograniczona pula adresów IPv4 – konieczność stosowania translacji
- Jakie utrudnienia wprowadza NAT w komunikacji VoIP?
  - P2P
  - Komunikacja w stronę klienta

# Typy NATów (1)

**"Full Cone" NAT**

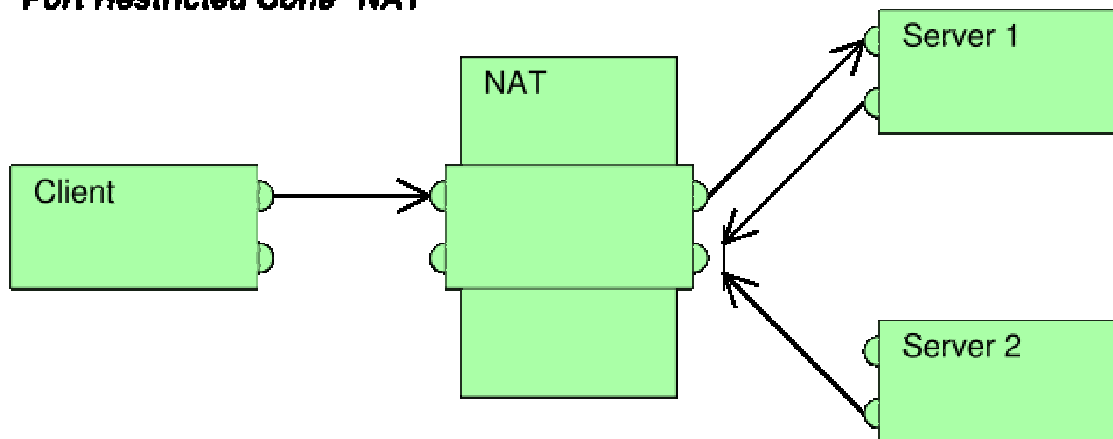


**"Restricted Cone" NAT**

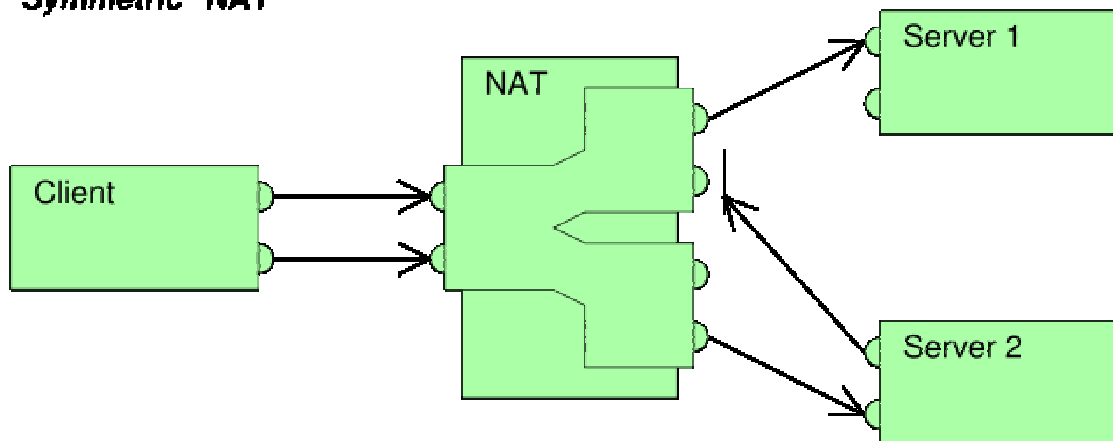


# Typy NATów (2)

**"Port Restricted Cone" NAT**



**"Symmetric" NAT**



---

# STUN (RFC 3489)

- **S**imple **T**raversal of **U**DP through **N**ATs
- Umożliwia klientowi uzyskanie informacji o NACie, za którym się znajduje
  - Typ NATa
  - Publiczny (NATowany) adres IP
- Nie umożliwia obejścia NATów symetrycznych, nie działa dla TCP
- Serwer STUN posiada dwa adresy IP
- Detekcja wg maszyny stanów

# Relaying

- **TURN (Traversal Using Relay NAT)**
  - Serwer pełni rolę proxy
  - Działa dla wszystkich typów NATów, także dla TCP
  - Spory koszt utrzymania serwera
- **Rozproszona sieć proxy**
  - Skype: ordinary host & super nodes
  - skomplikowane mechanizmy zarządzania siecią

---

## UPnP / IGD

- Internet **G**ateway **D**evice protocol
- Wchodzi w skład rodziny protokołów **U**niversal **P**lug and **P**lay
- Router/NAT wspierający IGD udostępnia klientowi:
  - Poznanie adresu zewnętrznego
  - Przeglądanie, dodawanie, usuwanie i modyfikacje wiązań NAT

---

## NAT – informacje dodatkowe

- Inne sposoby obejścia:
  - komunikacja w drugą stronę
  - komunikacja po sieci LAN, o ile to możliwe
- Czas życia wiązań
- Inteligentne NATy:
  - wnikanie w protokół i modyfikacje pakietów w locie
  - brak możliwości tworzenia wiązań dla sesji w ramach LAN

# Protokoły - plan

- IAX2
- SIP/SDP/RTP

# IAX2

- **Inter-Asterisk Exchange**, wersja 2
- Stworzony w celu łączenia internetowych centralk PBX Asterisk
- Podstawowe cele:
  - Minimalizacja pasma
  - Redukcja problemów z NATami
- Dwa protokoły w jednym: sygnalizacja i dane

# IAX2

- Trzy rodzaje ramek
  - Full frame
  - Mini frame
  - Meta frame
- Wsparcie dla szyfrowania (AES)
- IAX2 może przenosić oprócz dźwięku także wideo, tekst, obraz
- Popularna implementacja: iaxClient

---

## IAX2 – zalety i wady

- Zalety:
  - Sygnalizacja i dane w jednym strumieniu UDP
  - Trunking
  - Wykorzystanie jednego portu
  - Protokół binarny
- Wady:
  - Dane przechodzą zawsze przez serwer
  - Prosty protokół (mniejszy zakres zastosowań)
  - Brak standaryzacji (draft, nie RFC)

---

# IP Multimedia Subsystem

- Session Initiation Protocol (RFC 3261-4)
- Session Description Protocol (RFC 4556)
- Realtime Transport Protocol (RFC 3550)
  
- Wielkie firmy telekomunikacyjne
- Standardyzacja IETF w formie RFC
- Rozwiązania praktyczne
- Podstawa telefonii 3G

---

## SIP: główne cechy

- Protokół tekstowy
- Transport UDP
- Podstawowa funkcjonalność: rejestracja i nawiązanie połączenia
- Rozszerzenie SIMPLE: instant messaging & presence
- Łączność z za NAT-a

---

# SIP: protokół tekstowy

- Format wiadomości jak w HTTP, obsługa MIME
- Istotne różnice
  - UTF-8
  - Kolejność nagłówek
- Szyfrowanie S/MIME – tylko payload!
- Format tekstowy – łatwy do debugowania
- Bezpieczeństwo!

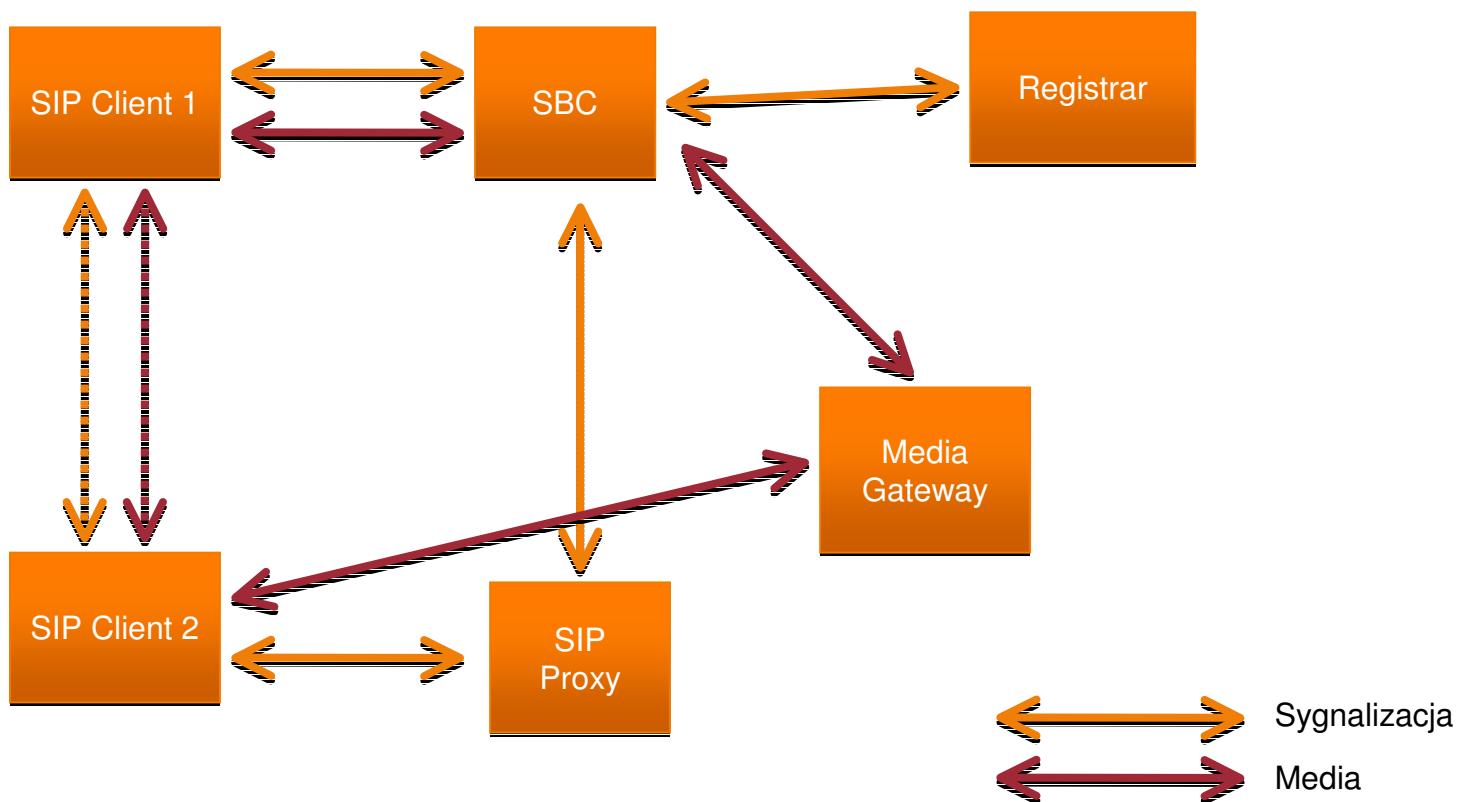
---

# SIP: Transport UDP

- Brak sesji – większa wydajność i skalowalność
- Stateless proxy
- Straty – wymagana rozbudowana warstwa transakcyjna
- Alternatywne transporty: TCP i TLS
- W przyszłości: DTLS

# SIP: Architektura

w wielkim uproszczeniu ;)



# SIP: Podstawowa funkcjonalność

- Rejestracja – związanie aktualnego adresu Internetowego klienta (kontaktu) z jego publicznym ID (SIP URL)
- Nawiazanie połączenia (sesji) – metoda INVITE

---

# SIP: rozszerzenie SIMPLE

- Komunikator internetowy
- Przesyłanie krótkich wiadomości tekstowych – metoda MESSAGE
- Publikowanie informacji o dostępności/statusie użytkownika (presence)
- Bramka SIP/XMPP

---

## SIP: Łączenie z za NAT-a

- Nagłówek Via
- Routing symetryczny: rport
- Mechanizmy keep-alive
- Wykorzystanie STUN/TURN
- „Zbyt mądre NAT-y”

---

# Opis sesji w SDP

- Wymiana informacji o parametrach sesji
  - Typ mediów (audio/video)
  - Transport (RTP)
  - Adresy (IP i port RTP)
  - Obsługiwane kodeki (payload map)
  - Kierunek transmisji

---

## SDP: negocjacja kodeka

1. Inicjator przysyła listę obsługiwanych kodeków w SIP INVITE – priorytet: kolejność na liście
2. Odbiorca ustala wspólny podzbiór i odsyła zmodyfikowany payload map w odpowiedzi

---

## SDP: modyfikacja parametrów sesji

- Powtórna wiadomość SIP INVITE ze zmodyfikowanym opisem SDP
- Zastosowanie: funkcja HOLD
  - Zmiana adresu IP RTP na 0.0.0.0
  - Zmiana kierunku sesji – „music on hold”

---

# Transport mediów: RTP

- Podstawowa funkcjonalność RTP
- Transmisja kodów DTMF
- Kodowanie szumu komfortowego
- RTP via NAT

---

# RTP: podstawowa funkcjonalność

- Porządkowanie pakietów
- Wykrywanie straty pakietu
- Identyfikacja nadawców w miksie
- Identyfikacja ładunku

---

# RTP: transmisja kodów DTMF

- Zastosowanie: automaty IVR i inne aplikacje telefoniczne
- In-band signalling
- Kodek telephone-event (RFC 2833)

## RTP via NAT

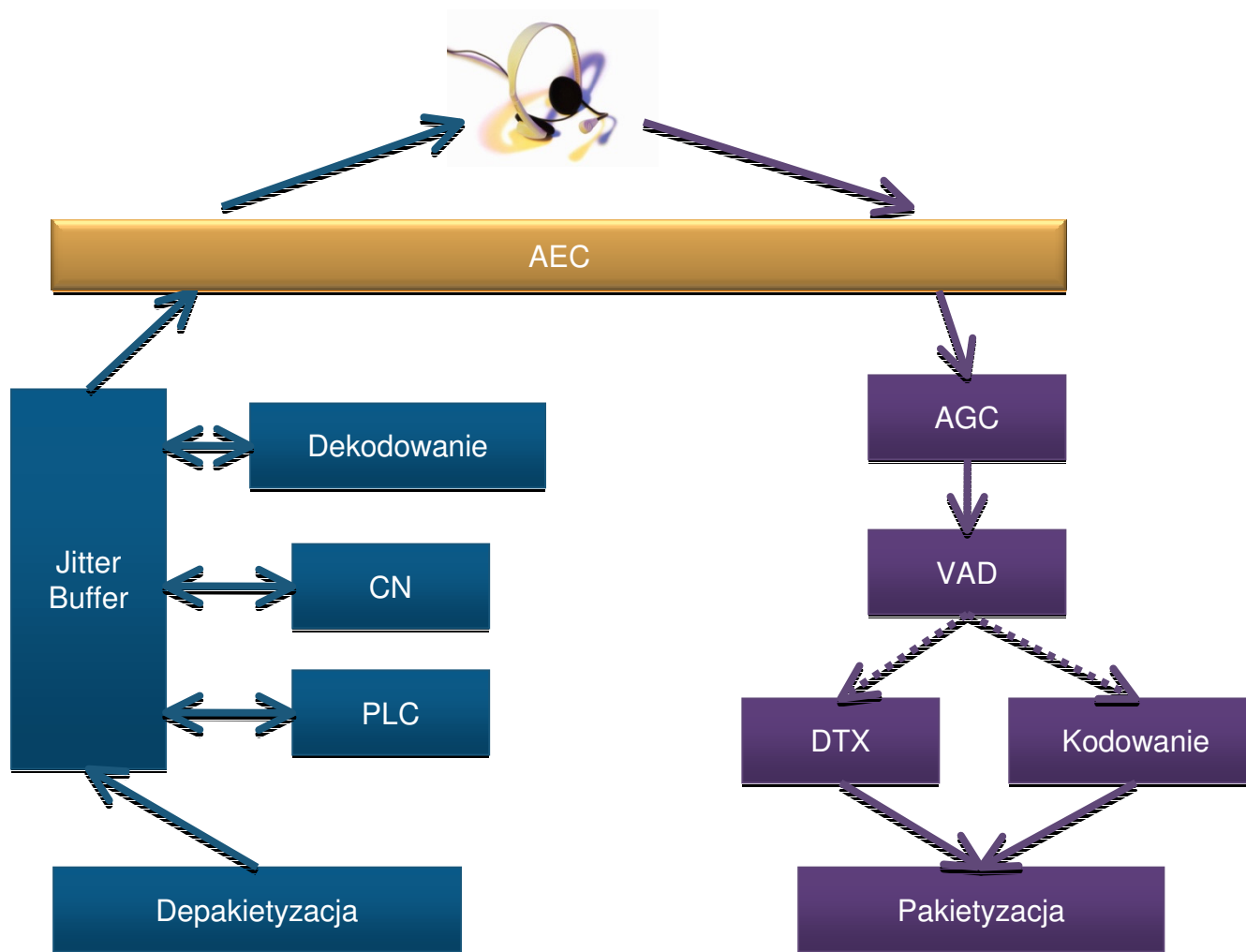
- Mechanizm early media/address & port latching
- Bezpieczeństwo!
- Metodologia ICE (draft-ietf-mmusic-ice-13)

---

# Transmisja mediów w VoIP

- Komponenty
- Kodeki dźwięku
- Czynniki pogarszające jakość dźwięku
- Mechanizmy QoS

# Komponenty systemu

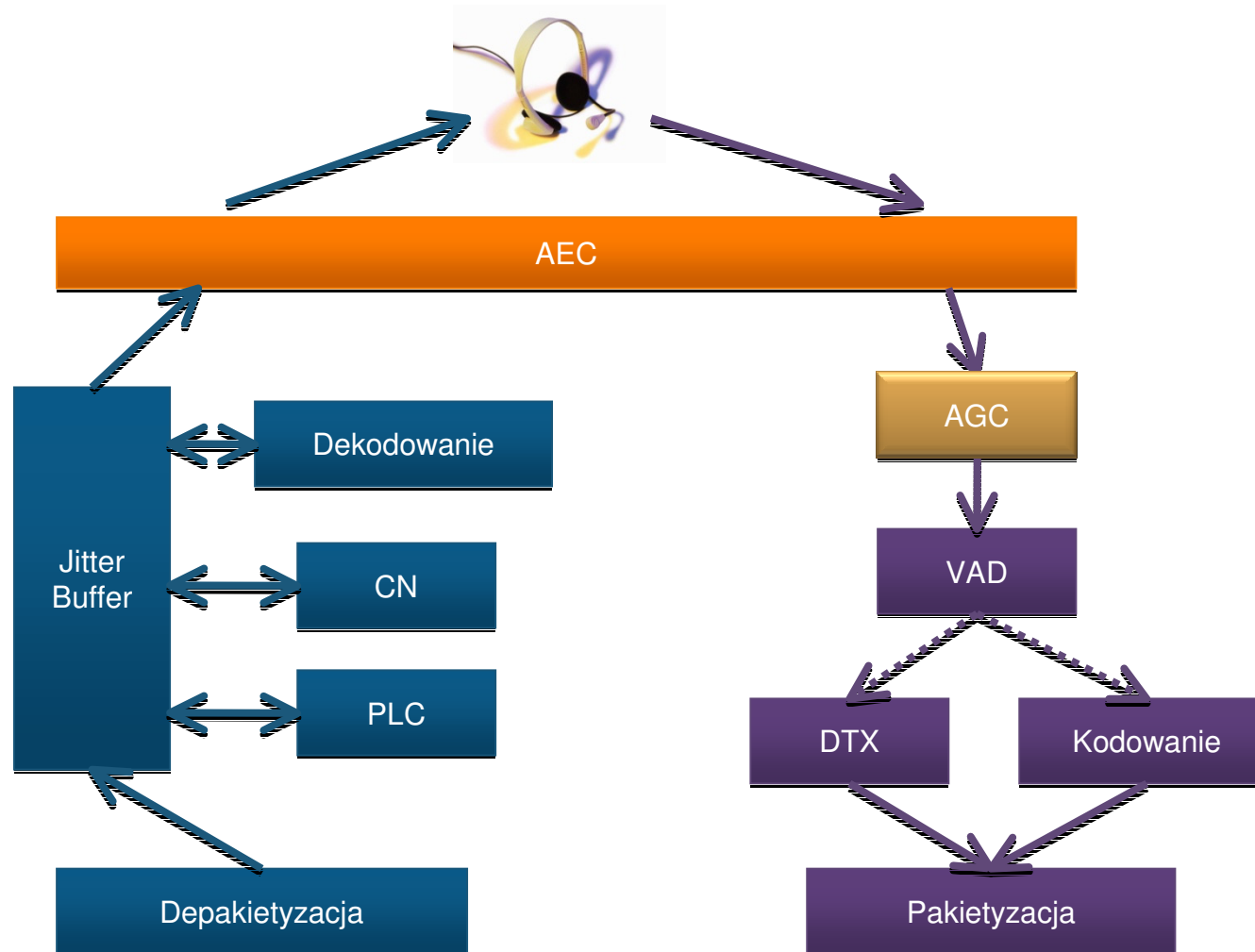


---

# Acoustic Echo Cancellation

- Przyczyny powstawania echa
- Czas echa 200-500ms
- Echo wielokrotne
- Algorytmy AEC
- Paradoks „tu włączam – tam efekt”

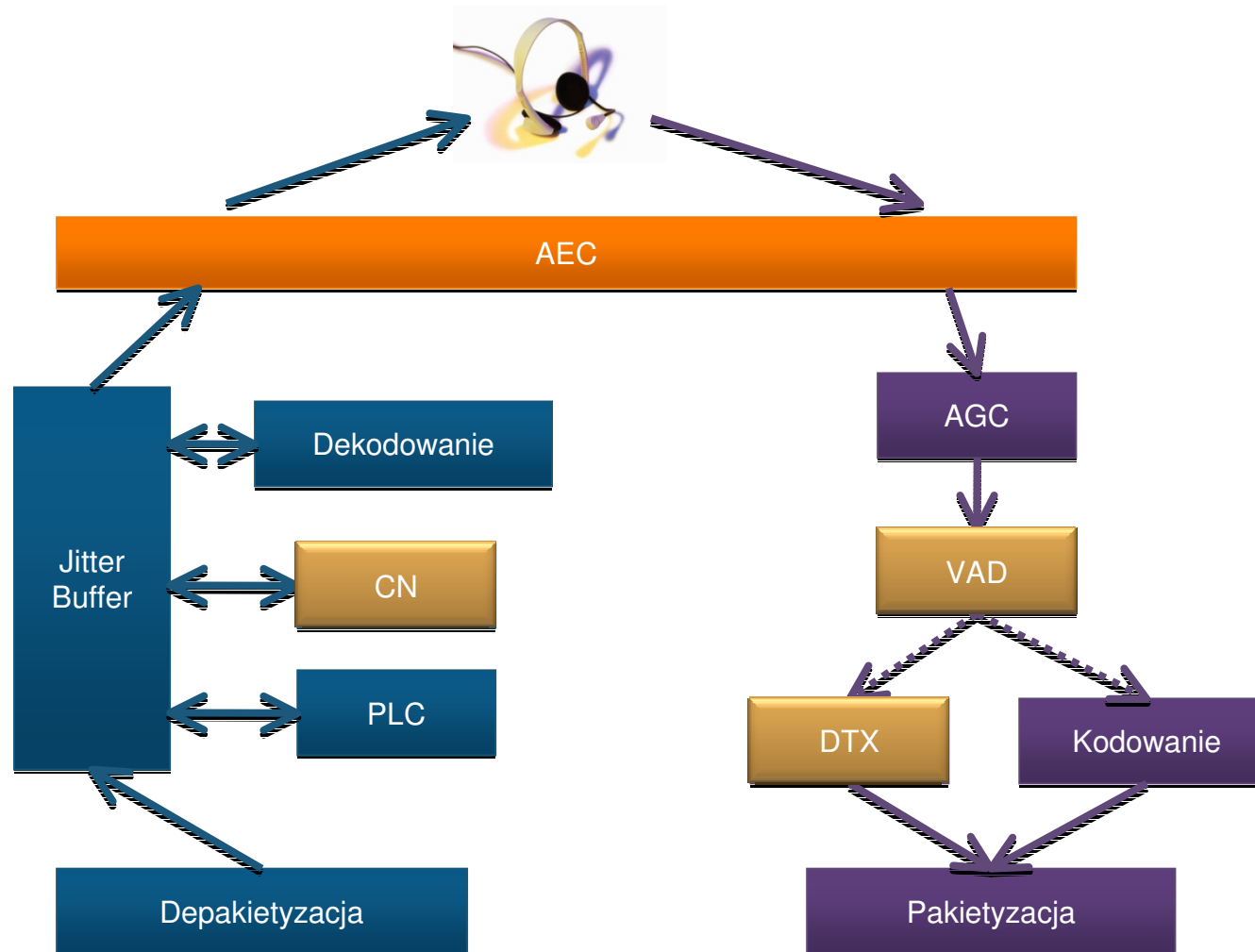
# Komponenty systemu



# Automatic Gain Control

- Problem „dalekiego mikrofonu”
- Algorytm AGC
- Zakłócenia
  - „Szum i łomot”
  - Sprzężenia

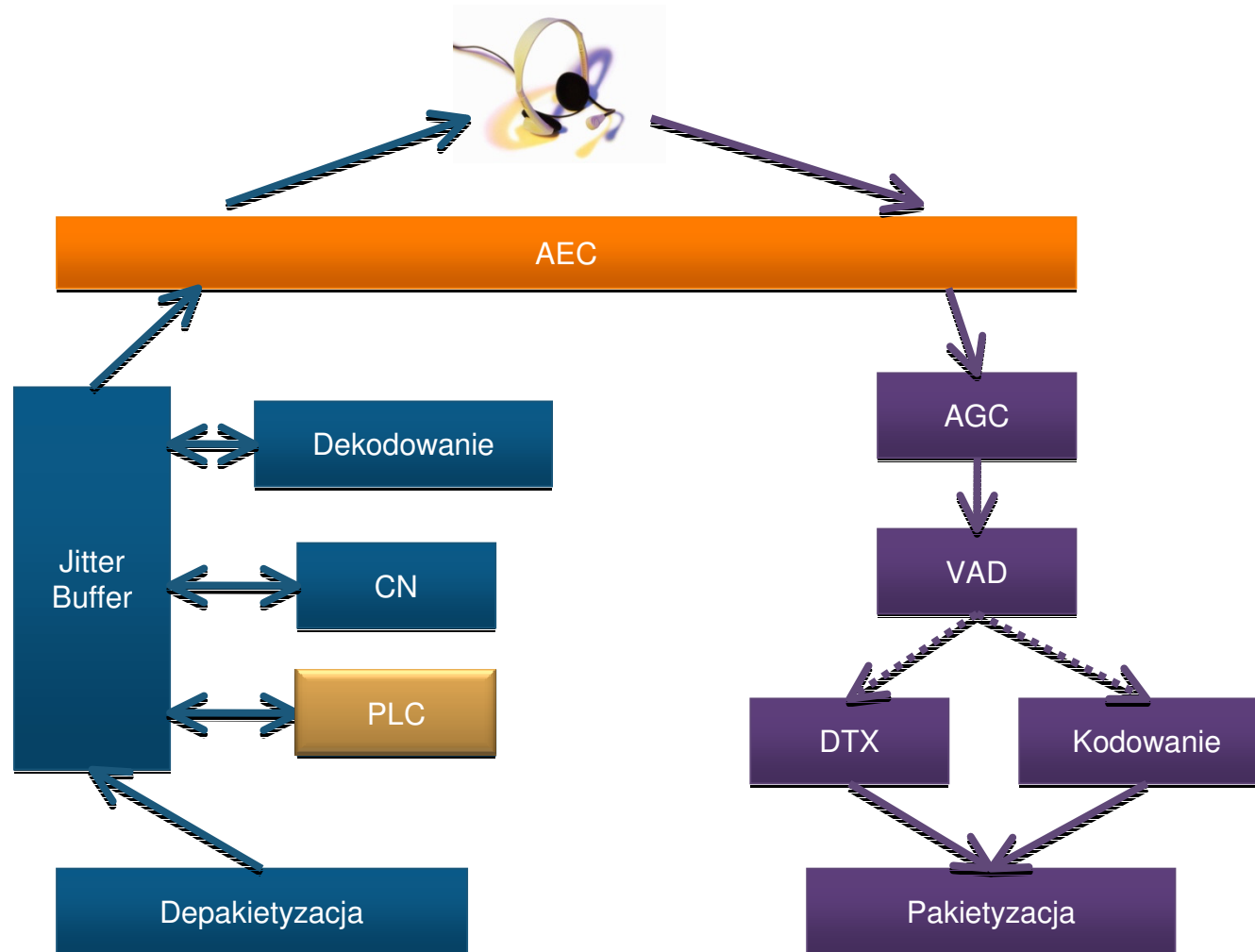
# Komponenty systemu



## VAD/DTX/CN

- Voice Activity Detection: jak wykryć ciszę?
- Discontinuous Transmission: ograniczenie obciążenia sieci
- Comfort Noise: „halo? jesteś tam?”
- Wsparcie ze strony kodeka

# Komponenty systemu

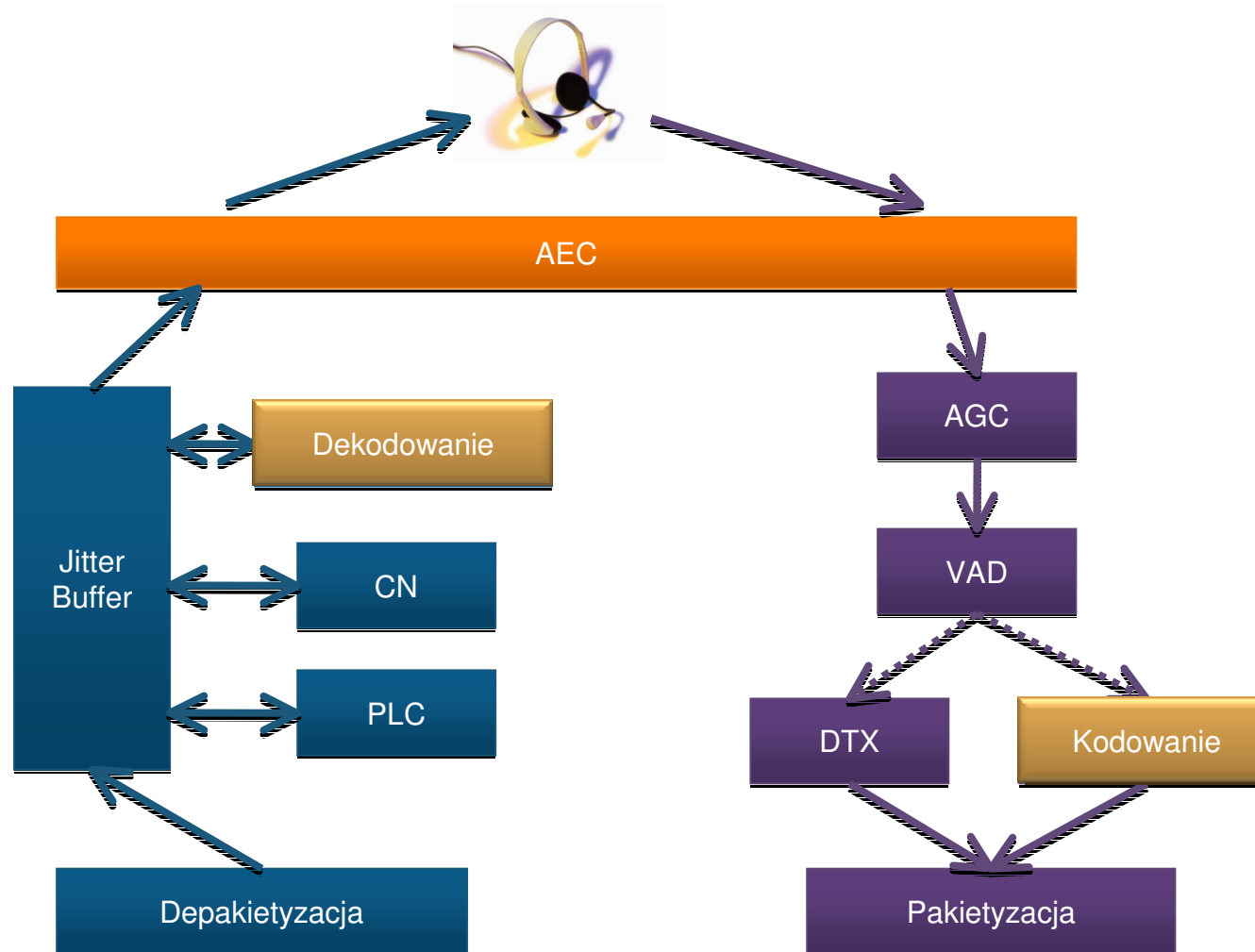


---

# Packet Loss Concealment

- Typowe implementacje
  - Powtórzenie pakietu
  - Zastąpienie szumem
  - Powtórzenie z przemiksowaniem z szumem
  - Predykcja liniowa
- Wsparcie ze strony kodeka
- Wybór algorytmu PLC

# Komponenty systemu



# Kodeki dźwięku

Kodek (CODEC): Koder/Dekoder;  
algorytm kodowania (kompresji) dźwięku

- Kryteria wyboru kodeka
- Ocena jakości dźwięku kodowanego

---

# Kodek: kryteria wyboru

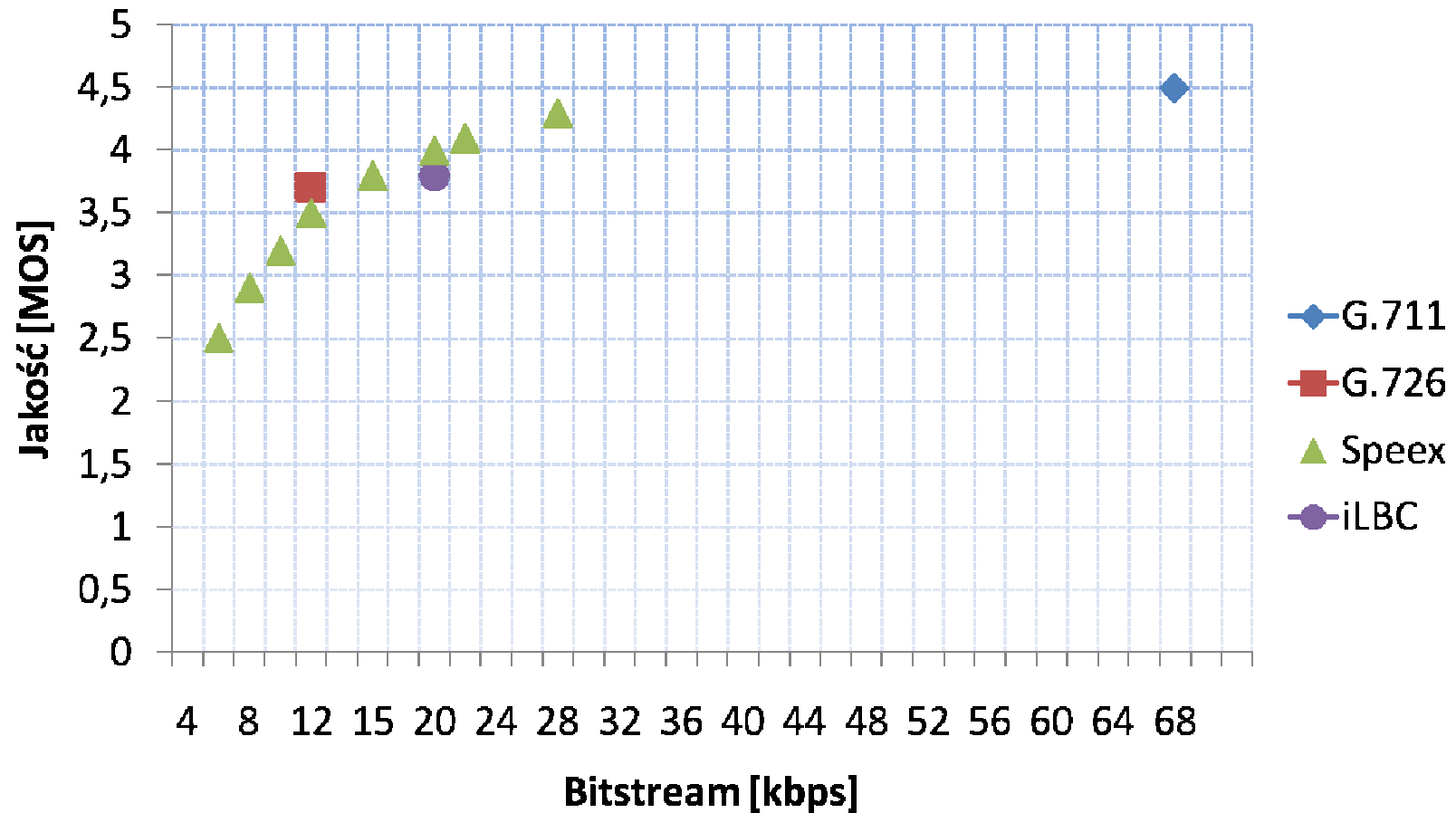
- Złożoność obliczeniowa
- Stopień kompresji
  - Prosty kodek – niska kompresja (G.711 64kbps)
  - Złożony kodek – wysoka kompresja (G.729 8kbps)
- Jakość dźwięku
- Licencja!

---

## Kodek: ocena jakości

- Skala MOS (Mean Opinion Score) –  
wynik testów odsłuchowych  
(subiektywnych)
- Symulacja MOS via PESQ (Perceptual  
Evaluation of Speech Quality ITU-T  
P.862)
  - Miara obiektywna
  - Wady!

# Kodek: jakość/pasmo



---

# Czynniki pogarszające jakość

- Kompresja
- Straty pakietów
- Opóźnienie
- Jitter

# Kompresja

- Pamiętamy wykres „narzut protokołu”?
- Zbyt wysoka kompresja – utrata jakości bez zysku
- Optymalny bitrate w sieci IPv4 ~16-20kbps

## Straty pakietów

- Typowy transport – UDP
- Utrata pojedynczego pakietu - z reguły niezauważalna przy dobrym algorytmie PLC
- Straty sekwencyjne

---

## Opóźnienie

- „Czas niekomfortowy” – według standardów telekomunikacyjnych 35ms
- Typowo w VoIP 200-300ms
- Przyczyny
  - Buforowanie w karcie dźwiękowej (40-100ms)
  - Bezładność (latency) systemu operacyjnego
  - Buforowanie w sieci

# Jitter

- Opóźnienie zmienne w czasie
- Przyczyny
  - Duży kwant czasu procesora (rzędu długości ramki audio)
  - Zmienne obciążenie routerów
  - Dryft zegara

---

# Jitter: przeciwdziałanie

- Buforowanie
- Adaptacja długości bufora
  - Synteza ramek
  - Sklejanie ramek
  - Związek z PLC
  - Optymalny moment adaptacji?

# Mechanizmy QoS

- Znakowanie pakietów IP
- Protokół RSVP

---

## QoS: znakowanie pakietów

- Pole Type of Service w nagłówku IP
- Mała ziarnistość kontroli
- Niejednoznaczna interpretacja
- Bardzo często ignorowane
- Windows XP i nowsze –  
niemodyfikowalne bez uprawnień  
administratora, „silently ignored”

## QoS: RSVP

- Resource Reservation Protocol – rezerwacja zasobów sieciowych
- Precyzyjna kontrola – min., max. pasmo, żądane gwarancje
- Traffic shaper wymusza interwał nadawania
- Stosowanie w Internecie mocno wątpliwe – tylko sieci pod kontrolą

Dziękujemy!  
Pytania?