

Wielosługowa sieć IP QoS: architektura i praktyczna weryfikacja w instalacji pilotowej

Od wielu lat dążeniem projektantów sieci telekomunikacyjnych, jak również projektantów sieci komputerowych, jest stworzenie takiej infrastruktury technicznej, która umożliwiłaby efektywnie przekazywać informację związaną z różnorodnymi klasami aplikacji (usługami), tj. opartymi na przekazie mowy, danych, obrazów ruchomych (video) i nieruchomych. Sieć spełniająca powyższe wymagania nazywamy siecią wielosługową (*multi-service*). Zasadniczą trudnością przy opracowaniu właściwej techniki telekomunikacyjnej dla sieci wielosługowej jest zagwarantowanie należytej jakości przekazu informacji przez sieć dla każdej z oferowanych klas aplikacji. Ogólnie ujmując, dostępne klasy aplikacji różnią się pomiędzy sobą zarówno typem (profilem) ruchu generowanego do sieci, wymaganą przepływnością bitową, jak i wymaganiami dotyczącymi jakości przekazu. Przykładowo, przekaz mowy wymaga przepływności, w zależności od użytej metody kodowania w granicach od rzędu kilku kbit/s do 64 kbit/s, z jednoczesnym zapewnieniem małych i stałych opóźnień, maksymalnie do 150 ms (w jednym kierunku), przy stosunkowo dużym dopuszczalnym poziomie strat, nawet do 10^{-2} . Z drugiej strony np. przekaz zbioru danych wymaga efektywnej przepływności rzędu setek kbit/s i więcej, przy zerowym końcowym poziomie strat. W tym przypadku opóźnienie przekazu tego zbioru jest parametrem drugorzędym. Jest zatem oczywiste, że spełnienie w sieci wielosługowej tak przeciwstawnych wymagań dotyczących jakości przekazu informacji, przy użyciu tej samej infrastruktury, wymaga zarówno nowych rozwiązań architektonicznych, jak i technik telekomunikacyjnych. Ostatecznie wymaga się, aby wykorzystanie zasobów sieci było jak największe, co zapewniają jedynie efektywne metody multipleksacji.

* Instytut Telekomunikacji Politechniki Warszawskiej, e-mail: wojtek@tele.pw.edu.pl

** Telekomunikacja Polska S.A, Centrum Badawczo-Rozwojowe, Warszawa oraz Instytut Telekomunikacji Politechniki Warszawskiej, e-mail: marek.dabrowski@mail.tpsa.pl

Podsumowując powyższe należy stwierdzić, że zasadniczym wymaganiem wobec sieci wielousługowej jest zapewnienie przekazu informacji na poziomie wymaganym dla każdej z oferowanych w sieci aplikacji. Sieć taka powinna być publiczna, co oznacza iż koszt jej instalacji i utrzymania powinien być ekonomicznie uzasadniony.

Obecnie w zasadzie istnieją dwa rodzaje sieci: realizująca głównie usługę przekazu mowy i ukierunkowana na przekaz danych. Pierwsza z tych sieci jest oparta na technice komutacji kanałów, druga zaś na technice komutacji pakietów. Zatem zasadniczym problemem przy realizacji sieci wielousługowej jest zapewnienie przekazu mowy w sieci z komutacją pakietów lub zapewnienie przekazu danych w sieci z komutacją kanałów. Do realizacji sieci wielousługowej opracowano wiele technik, w tym technikę **N-ISDN** (*Narrowband Integrated Services Digital Network*) opartą na komutacji kanałów oraz technikę **ATM** (*Asynchronous Transfer Mode*) opartą na komutacji pakietów. Obie te techniki, w swoich założeniach, mają pewne mechanizmy służące zapewnieniu odpowiedniej jakości obsługi dla każdej z klas aplikacji. Należy dodać, iż prace badawcze i standaryzacyjne dla tych technik trwały wiele lat i obecnie można je uważać na zakończone (lub, w przypadku ATM, będące na ukończeniu).

Faktem jest, iż obecnie najbardziej rozwiniętą siecią, mającą pewne cechy sieci wielousługowej, jest globalna sieć Internet oparta na technice komutacji pakietów z zastosowaniem protokołu **IP** (*Internet Protocol*). Liczba abonentów sieci Internet jest obecnie porównywalna z liczbą abonentów sieci telefonicznej, zaś ruch przenoszony w tej sieci w niektórych krajach przekroczył ruch telefoniczny i stale wzrasta. Sieć Internet obecnie nie zapewnia wprawdzie wymaganej jakości przekazu, jednak rekompensatą może być atrakcyjność i bogactwo dostępnych w sieci aplikacji i mały koszt opłat. Obecnie w sieci Internet jest dostępny cały zestaw aplikacji, z których najpopularniejsze to poczta elektroniczna (*e-mail*), **WWW**, **FTP** (*File Transfer Protocol*), Telnet, e-commerce (usługi bankowe, zakupy, rezerwacje hoteli, biletów itd.). W ostatnim czasie obserwuje się duże zainteresowanie wprowadzeniem do sieci Internet usługi przekazu mowy **VoIP** (*Voice over IP*). Rozpowszechnienie jej może doprowadzić w konsekwencji do „przejęcia” realizacji rozmów telefonicznych przez sieć Internet. Biorąc to pod uwagę, sieć Internet należy obecnie traktować jako poważnego kandydata do przekształcenia się w sieć wielousługową, oferującą wszystkie dostępne obecnie aplikacje, w tym również usługę telefonii. Obecnie czynnikiem ograniczającym atrakcyjność Internetu jest to, że nie zapewnia ona należytej jakości przekazu informacji. Dlatego też jest niezbędne wprowadzenie do sieci Internet dodatkowych elementów zapewniających wymaganą jakość obsługi, tj. odpowiedniej architektury sieci i skojarzonych z nią tzw. mechanizmów **QoS** (*Quality of Service*). Obecnie prace nad

modyfikacją sieci Internet są realizowane w większości ośrodków badawczych pracujących na rzecz telekomunikacji, również w ramach projektów międzynarodowych i ciał normalizacyjnych, w tym przypadku w ramach **IETF** (*Internet Engineering Task Force*) i **ITU-T** (*International Telecommunications Union*).

W artykule są przedstawione wyniki potwierdzające możliwości zapewnienia zróżnicowanej jakości przekazu informacji w sieci opartej na protokole IP w wersji 4. Rozważana koncepcja sieci opiera się na architekturze **DiffServ** (*Differentiated Services*) wzbogaconej o dodatkowe mechanizmy gospodarowania zasobami sieci i sterowania ruchem. Omawiane podejście jest obecnie wdrażane w formie prototypu, w ramach projektu europejskiego AQUILA (5. Ramowy Program Unii Europejskiej), w którym uczestniczą m.in. autorzy artykułu. Przedstawione w artykule wyniki zostały uzyskane w laboratoryjnej instalacji sieci pilotowej w **Centrum Badawczo-Rozwojowym Telekomunikacji Polskiej S.A.** W szczególności dotyczą one możliwości oferowania w sieci IP aplikacji wykorzystujących przekaz mowy i wideo. Ponadto jest przedstawiona architektura całej sieci wraz z krótkim omówieniem zaimplementowanych mechanizmów zapewniających QoS.

1 TECHNIKI TELEKOMUNIKACYJNE DO REALIZACJI SIECI WIELOUSŁUGOWEJ

Cechą rozróżniającą techniki w sieciach telekomunikacyjnych i komputerowych jest użyta technika komutacji. Rozróżnia się dwie podstawowe techniki komutacji, czyli technikę komutacji kanałów i technikę komutacji pakietów.

Proponowanym rozwiązaniem realizacji sieci wielousługowej opartej na technice komutacji kanałów jest technika N-ISDN, rozwijana od końca lat 70. Technika ta umożliwia komutację kanałów o stałej szybkości $n \times 64$ kbit/s. Takie podejście zapewnia efektywny przekaz mowy, jednakże konieczność zestawiania połączeń o stałych i znormalizowanych szybkościach w istotny sposób ogranicza przekaz danych; w przypadku ruchu o zmiennej szybkości nadawania i jednoczesnej realizacji wielu połączeń krótkotrwałych przydział kanału o stałej szybkości dla każdego połączenia oddzielnie jest po prostu nieefektywny lub wręcz niemożliwy do realizacji np. w warunkach miliona jednoczesnych połączeń, co obserwuje się w obecnych sieciach danych. W konsekwencji technikę N-ISDN należy traktować jako technikę realizującą głównie przekaz mowy i w ograniczonym zakresie przekaz danych i wideo. Dodatkowo technika N-ISDN nie jest rozwijalna w tym sensie, iż nie zapewnia możliwości wprowadzenia nowych mechanizmów dla zróżnicowania jakości przekazu w sieci, co jest wymagane w przypadku sieci wielousługowej.

ATM jest techniką opartą na komutacji pakietów o stałej długości (53 bajty), zwanych komórkami. Rozwój tej techniki w latach 90. doprowadził do osiągnięcia takiego poziomu, iż obecnie technika ta oferuje sześć usług sieciowych tj. **CBR** (*Constant Bit Rate*), **rt-VBR** (*real-time Variable Bit Rate*), **nrt-VBR** (*non real-time Variable Bit Rate*), **ABR** (*Available Bit Rate*), **GFR** (*Guaranteed Frame Rate*) i **UBR** (*Unspecified Bit Rate*)¹. Każda z nich jest przystosowana do efektywnego przekazu innego rodzaju ruchu z odmiennymi wymaganiami. Przykładowo, usługa CBR zapewnia przekaz danych o stałej szybkości (emulacja łącza), co umożliwia przekaz mowy z jakością porównywalną do oferowanej w sieci z komutacją kanałów. Obecnie sieci ATM są głównie realizowane w sieci szkieletowej do integracji różnych rodzajów ruchu (tj. telefonii i danych). Głównym czynnikiem hamującym rozwój ATM jest brak na rynku aplikacji bezpośrednio korzystających z tej techniki. Przykładem jest usługa ABR, zaprojektowana w celu efektywnego przekazu danych komputerowych, z której obecnie nie może korzystać żadna aplikacja (brak zaimplementowanych odpowiednich mechanizmów w aplikacjach). Należy jednakże podkreślić, iż mechanizmy zastosowane w technice ATM stanowią podstawę dla mechanizmów przewidywanych w sieci IP z jakością obsługi.

Technika IP jest oparta na komutacji pakietów o zmiennej długości. W sieci Internet, opartej na protokole IP v.4, oferuje się przekaz danych według zasady *best effort*. Oznacza to, iż dostęp do sieci nowych strumieni ruchu nie jest ograniczany przez mechanizmy sieciowe zaś sterowanie ruchem w ramach danego połączenia odbywa się za pomocą protokołów **TCP** (*Transmission Control Protocol*). W konsekwencji przekaz danych odbywa się bez zapewnienia gwarancji dotyczących takich parametrów, jak prawdopodobieństwo utraty pakietu, dopuszczalny czas przekazu pakietu przez sieć, zmienność (*jitter*) opóźnienia przekazu itp. Powyższy brak gwarancji uniemożliwia efektywny przekaz informacji związanych z różnymi aplikacjami, co jest wymagane w sieci wielousługowej. Dlatego też jest konieczne wprowadzenie do sieci nowych mechanizmów zapewniających jakość obsługi, tzw. mechanizmów QoS.

2 WIELOUSŁUGOWA SIEĆ IP QoS

¹ CBR – usługa stałej szybkości bitowej, rt-VBR – zmiennej szybkości bitowej czasu rzeczywistego, nrt-VBR – zmiennej szybkości bitowej, ABR –osiągalnej szybkości bitowej, GFR – gwarantowanej szybkości przekazu ramek, UBR – nieokreślonej szybkości bitowej

Zapewnienie wielosługowego charakteru sieci IP wymaga wprowadzenia zróżnicowanych usług sieciowych, podobnie jak w przypadku sieci ATM. Liczba tych usług nie powinna być za duża (np. 4 usługi), a poszczególne usługi powinny różnić się pomiędzy sobą profilem przekazywanego ruchu (o stałej czy o zmiennej szybkości) oraz poziomem gwarantowanego QoS, reprezentowanego przez wartości dopuszczalnych opóźnień przekazu pakietów wraz z ich zmiennością (*jitter*), prawdopodobieństwem utraty pakietów wynikającym z chwilowych przeciążeń sieci itd.

W sieciach wielosługowych wyróżnia się dwa podstawowe rodzaje ruchu, tj. ruch **strumieniowy** (*stream*) i ruch **elastyczny** (*elastic*). Ruch strumieniowy jest generowany do sieci przez takie aplikacje, jak przekaz obrazów ruchomych (np. *video*) czy dźwięku (*audio*, *Voice over IP*). Ruch ten powinien być przesyłany przez sieć z małym (pomijalnym) opóźnieniem przy jednoczesnym zapewnieniu niskich strat. Ruch elastyczny dotyczy przesyłania takich dokumentów, jak zbiory, obrazy nieruchome itd. Dla tego typu ruchu wymaga się zapewnienia poprawności przekazu, natomiast wymagania dotyczące czasu tego przekazu nie są zazwyczaj krytyczne. W tabeli 1 przedstawiono wymagania dotyczące jakości przekazu pakietów przez sieć dla wybranych usług multimedialnych.

Tabela 1: Wymagania dotyczące jakości przekazu dla wybranych usług multimedialnych

Usługi multimedialne	Wrażliwość na:		Przykłady
	straty pakietów	opóźnienia	
Interaktywne wideo: czasu rzeczywistego małej rozdzielczości	duża duża	duża średnia	wideokonferencja <i>CU-SeeMe</i> (PC komputer)
Odtwarzanie wideo	duża	mała	wideo na żądanie wideo ze strony Web
Stały obraz	duża	mała	obraz ze strony Web
Faks	mała	mała	zdalne kopiowanie
Interaktywna mowa Odtwarzanie mowy	mała mała	duża mała	telefonია mowa ze strony Web
Interaktywne dane: dużej szybkości małej szybkości Nie-interaktywne dane	duża duża duża	duża średnia mała	sterowanie w czasie rzeczywistym Telnet E-mail

Należy wspomnieć, iż często jako argument dla zapewnienia wielosługowego charakteru sieci IP jest przytaczany fakt szybszego zwiększania dostępnych możliwości obsługi ruchu przez sieć (tj. przepływności łączy transmisyjnych i szybkości przetwarzania pakietów w węzłach) niż wzrost ruchu oferowanego do sieci. Idąc dalej tym tokiem rozumowania, nie należy zatem specjalnie przywiązywać wagi do zapewnienia przekazywanym pakietom odpowiedniej jakości, sama sieć bowiem nie jest nigdy przeciążana. Niestety, jak wskazują

doświadczenia użytkowników i operatorów sieci, przeciążenie sieci Internet jest nagminne i trwa prawie całą dobę.

W obecnych sieciach IP wykorzystuje się podejście *best effort* w celu przekazu pakietów do miejsc przeznaczenia. Oznacza to, że routery IP przesyłają pakiety w kolejności „pierwszy przyszedł - pierwszy obsłużony” oraz że w przypadku wypełnienia buforu, po prostu odrzucają pakiety nadmiarowe, bez względu na ich ważność. Takie rozwiązanie jest oczywiście niedopuszczalne np. dla przekazu mowy.

Pod terminem mechanizmy QoS należy rozumieć odpowiednie oprogramowanie i rozwiązania sprzętowe, które umożliwiają identyfikację, klasyfikację i przekaz pakietów należących do danego strumienia zgodnie z narzuconymi wymaganiami dotyczącymi jakości ich przekazu przez sieć. Ogólnie realizacja QoS wymaga m.in. wprowadzenia w routerach IP wielu kolejek i odpowiednich algorytmów do ich obsługi.

2.1 Mechanizmy QoS

Aby sieć telekomunikacyjna realizowała usługi z gwarantowaną jakością, czyli spełniała wymagania QoS, należy w niej zaimplementować odpowiednie mechanizmy. Należy podkreślić, iż mechanizmy te są odmienne w sieciach z komutacją kanałów i w sieciach z komutacją pakietów. Dla przypomnienia - w sieciach z komutacją kanałów obsługujących ruch telefoniczny parametrem obrazującym jakość sieci jest prawdopodobieństwo odrzucenia wywołania. Bardziej skomplikowany charakter opisu parametrów QoS występuje w sieciach pakietowych, a więc opartych na takich technikach, jak ATM czy IP. Ogólnie w sieciach tych rozróżnia się dwa zasadnicze poziomy, tj.:

- poziom wywołań,
- poziom pakietów IP (komórek ATM).

Wymagania dotyczące QoS na poziomie wywołań są takie same jak w przypadku sieci z komutacją kanałów i wyrażają się zazwyczaj podaniem wartości prawdopodobieństwa odrzucenia zgłoszenia, zwykle przyjmowanego na poziomie 0,01. Inne są jednakże warunki, przy których to wymaganie powinno być spełnione. Otóż w sieci telefonicznej operuje się pojęciem **GNR** (godzina największego ruchu), natomiast w sieciach pakietowych (zwłaszcza w sieci Internet) obciążenie sieci jest mniej zróżnicowane w ciągu doby.

Drugi poziom dotyczy jakości przekazu pakietów i odnosi się do takich parametrów, jak opóźnienie przekazu pakietu przez sieć, zmienność tego opóźnienia, prawdopodobieństwo utraty pakietu wynikłe z chwilowego przeciążenia sieci itd.

W konsekwencji, w celu zapewnienia wymaganych parametrów QoS, w sieci pakietowej należy wprowadzić funkcję przyjmowania nowych wywołań, która z kolei musi wziąć pod

uwagę charakter przesyłanego przez sieć ruchu na poziomie pakietów (komórek). Inaczej ujmując, nowe połączenie może być przyjęte do obsługi jedynie w przypadku, gdy przyjęcie to nie spowoduje pogorszenia jakości obsługi (na poziomie pakietów) już istniejących połączeń.

Zapewnienie odpowiedniej jakości przekazu przez sieć IP wymaga - podobnie jak w sieci ATM - zdefiniowania odpowiednich funkcji realizujących QoS. Postępując w tym duchu należy:

- zdefiniować klasy usług sieciowych wraz z mechanizmami zapewniającymi sterowanie ruchem; ogólnie mechanizmy takie mogą być zaimplementowane w każdym węźle (w tym przypadku w ruterze IP); przykładem usługi sieciowej jest usługa gwarantująca przekazanie pakietów przez sieć z możliwie małym opóźnieniem i z zapewnieniem minimalnych strat;
- określić zasoby sieci przynależne danej usłudze sieciowej; w przypadku sieci pakietowej należy dla danej usługi sieciowej przydzielić odpowiednią przepływność na każdym łączu wyjściowym, wraz z buforem dla gromadzenia pakietów powodujących chwilowe przeciążenia;
- w ruterze IP zaimplementować mechanizmy zapewniające rozróżnienie pakietów według ich przynależności do danej usługi sieciowej;
- zdefiniować dla każdej z usług sieciowych tzw. parametry QoS, określające jakość obsługi odbieraną na poziomie wywołania i na poziomie pakietów;
- zdefiniować funkcję realizującą przyjmowanie/odrzucając nowe wywołania;
- zdefiniować parametry połączenia, zgłaszane w fazie zestawiania połączenia i dotyczące charakterystyki oferowanego ruchu; zwykle są one podawane w formie parametrów tzw. mechanizmu *token-bucket*;
- nie dopuszczać do sieci ruchu, który nie jest zgodny z kontraktem ruchowym.

Jest więc oczywiste, że w sieci QoS IP powyższe mechanizmy muszą być zdefiniowane. Należy dodać, iż implementacja efektywnych mechanizmów QoS w sieci IP jest szczególnie trudna, co wynika z faktu, iż sieć ta nie realizuje połączeń wirtualnych (jak w przypadku sieci ATM) oraz długość przesyłanych pakietów jest zmienna. Świadczy to, iż w porównaniu z siecią ATM sterowanie jakością w sieci QoS IP będzie z pewnością mniej efektywne.

2.2 Proponowane architektury dla sieci IP z jakością obsługi

Sieć **TCP/IP** zakłada jedynie model usługi sieciowej *best effort* i zarządzanie ruchem jedynie na końcu systemu (*end system only traffic management*). Dla zapewnienia QoS, w ramach IETF zaproponowano dwie alternatywne architektury dla sieci IP, a więc:

- architekturę *Integrated Services* (z rezerwacją zasobów), oznaczaną w dalszej części jako architektura **Intserv**; zasoby sieci są w niej przydzielane dla danej aplikacji na żądanie,
- architekturę *Differentiated Services* (z priorytetyzowaniem strumieni ruchu), oznaczaną w dalszej części jako architektura **Diffserv**; strumienie ruchu są tu klasyfikowane zgodnie z uprzednio wprowadzonymi usługami sieciowymi i przesyłane w sieci z różnymi priorytetami.

2.2.1 Architektura Intserv

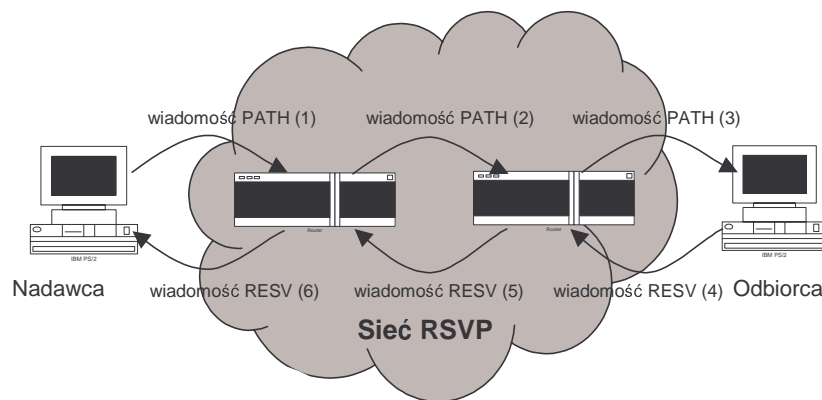
W architekturze IntServ [1] zakłada się, że zasoby w sieci są rezerwowane dla poszczególnych lub zagregowanych strumieni danych. W tym celu został zdefiniowany specjalny protokół sygnalizacyjny, znany pod nazwą protokołu **RSVP** (*Reservation Protocol*). Umożliwia on realizację żądania przez daną aplikację rezerwacji zasobów w sieci. Implementacja tego protokołu jest wymagana w każdym routerze IP. W konsekwencji router IP będzie przyjmował i realizował żądania rezerwacji, co wiąże się z przechowywaniem danych nt. każdej poczynionej rezerwacji wraz z informacją o skojarzonym strumieniu danych.

W architekturze IntServ, oprócz standardowej usługi typu *Best Effort*, zdefiniowano dwie dodatkowe usługi:

- **Guaranteed Service** – przeznaczoną dla aplikacji wymagających gwarancji odnośnie parametrów jakości przekazu danych związanych z opóźnieniami,
- **Controlled-load Service** – przeznaczoną dla aplikacji wymagających bezstratnego przekazu danych i charakteryzującą się jakością przekazu określaną jako lepsza niż *Best Effort*.

Zasadę działania protokołu RSVP ilustruje rys. 1. Żądanie rezerwacji zasobów jest inicjowane przez nadajnik (np. aplikację w terminalu końcowym). Wysyła on wiadomość typu *PATH message*, zawierającą informacje o charakterystyce generowanego ruchu tzw. *Traffic Specification (TSpec)*, tj. minimalne i maksymalne wartości żądanej przepływności, opóźnień przekazu pakietów przez sieć i ich zmienności. Wiadomość ta jest przekazywana do miejsca (lub wielu miejsc) przeznaczenia, przy czym następuje to od routera do routera, zgodnie z ustaloną przez algorytm drogą przez sieć. Odbiorca, po otrzymaniu wiadomości typu *PATH message*, wysyła wiadomość o rezerwacji typu *RESV message* w celu zarezerwowania

zasobów w sieci. Ponadto wiadomość *RESV message* przynosi informacje o typie żądanej usługi (**RSPEC**) oraz dotyczące identyfikacji pakietów należących do danego strumienia danych (*filter spec*), tj. numer portu, używany rodzaj protokołu transportowego itp. Na podstawie tych informacji routery mogą rozróżnić strumienie danych objęte rezerwacjami. Wiadomość *RESV message* jest przekazywana przez sieć w kierunku od ujścia do źródła. Każdy z routerów, przez który przechodzi ta wiadomość, realizuje funkcję przyjmowania wywołania, tj. autoryzacji i podejmowania decyzji o przydziale zasobów, czyli przydzielenia odpowiedniej przepływności łącza i wielkości bufora. Jeśli w danym routerze żądanie takie nie może być zrealizowane (np. z powodu braku dostatecznej ilości zasobów), wówczas router ten wysyła do odbiorcy wiadomość o odrzuceniu zgłoszenia. W przypadku akceptacji wywołania wiadomość *RESV message* jest przekazywana do następnego routera. Kiedy router końcowy (najbliższy źródła) otrzyma wiadomość *RESV message* i to wywołanie zaakceptuje, wówczas wysyła do odbiorcy potwierdzenie o zarezerwowaniu zasobów w sieci.



Rys.1. Przepływ wiadomości sygnalizacyjnych w protokole RSVP.

Podsumowując, protokół RSVP cechuje się następującymi własnościami:

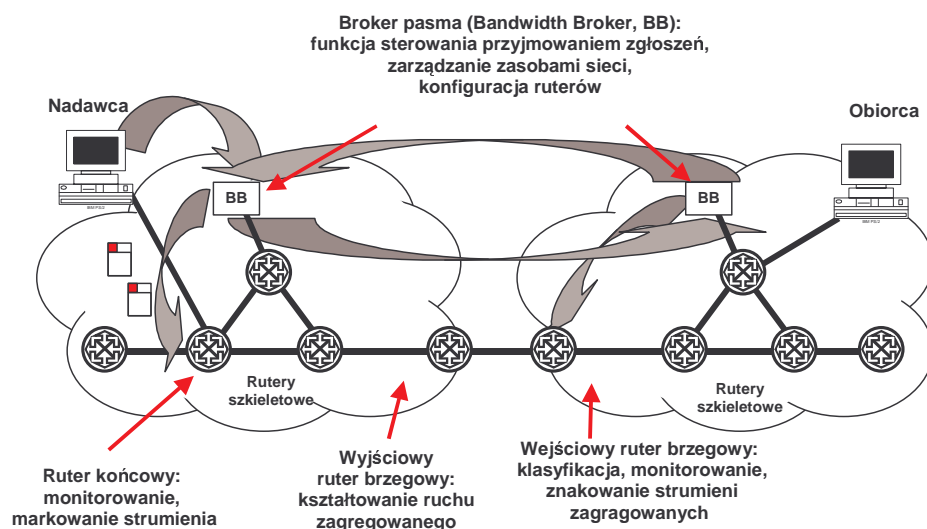
- jest to protokół sygnalizacyjny (a nie realizujący sterowanie przekazem danych),
- rezerwacja jest typu „soft”, tj. musi być odnawiana okresowo,
- żądanie rezerwacji jest generowane przez odbiorcę,
- użycie protokołu przez daną aplikację wymaga opracowania specjalnego interfejsu **API** (*Application Programming Interface*),
- wymaga przechowywania informacji w routerach o pojedynczych strumieniach danych, co w konsekwencji prowadzi do problemów ze skalowalnością sieci.

W architekturze *IntServ*, oprócz protokołu RSVP i algorytmu przyjmowania nowych strumieni danych, można wyróżnić jeszcze dwa mechanizmy, tj. klasyfikacji i kolejkowania pakietów. Pierwszy z nich dotyczy metod klasyfikacji pakietów na podstawie zdefiniowanych

pól w nagłówku pakietu. Pakiety pochodzące z różnych strumieni mogą należeć do tej samej klasy ruchu i są kierowane do tej samej kolejki. Drugi mechanizm dotyczy sposobu obsługi kolejek pakietów należących do różnych usług sieciowych, tak aby były spełnione wymagania dotyczące jakości obsługi.

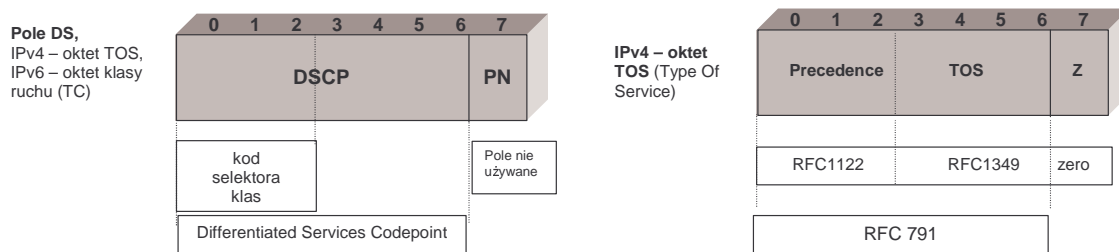
2.2.2 Architektura DiffServ

Architektura *DiffServ* [3, 4] (rys.2) została zaproponowana jako rozwiązanie, które omija problem skalowalności, występujący w architekturze *IntServ*. Założono w niej, że w sieci definiuje się *a priori* odpowiedni zestaw usług sieciowych jedynie na podstawie mechanizmów priorytetyzowania strumieni w routerach.



Rys.2. Ogólna architektura DiffServ

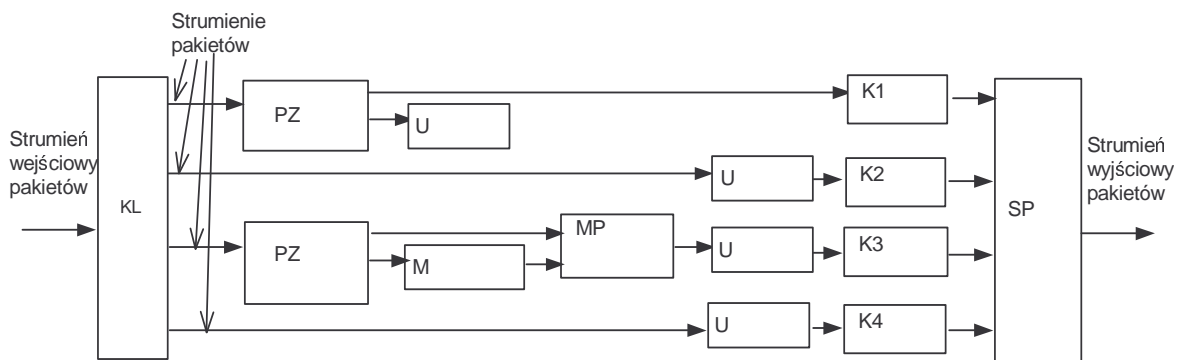
Klasyfikacji pakietów do poszczególnych usług sieciowych dokonuje się przez informacje zawarte w polu **TOS** (*Type Of Service*), występujące w nagłówku pakietu IPv4 [6] (rys.3). W architekturze tej zdefiniowano sposób wykorzystania pola TOS i określono zestaw reguł przekazywania pakietów w routerze tzw. **PHB** (*Per Hop Behaviours*).



Rys.3. Format oktetu IPv4 TOS i pola DS (Differentiated Services).

Dotychczas w opracowaniach IETF zdefiniowano dwie podstawowe zasady przekazu pakietów (PHB), które w najprostszym przypadku mogą reprezentować dwa poziomy obsługi pakietów:

- *Expedited Forwarding (EF)* jest określony przez pojedynczą wartość pola **DSCP** (*Differentiated Services Codepoint*) i wykorzystywany do zapewnienia jakości obsługi związanej z parametrami opóźnień. Ruch ten jest monitorowany na wejściu do sieci; pakiety nie spełniające warunków zawartych w profilu ruchowym danego strumienia lub grupy strumieni są usuwane z sieci.
- *Assured Forwarding (AF)* określa cztery klasy i trzy poziomy odrzucania pakietów wewnątrz każdej z klas (w sumie 12 wartości pola DSCP). Ruch nie spełniający warunków zawartych w profilu ruchowym danego strumienia lub grupy strumieni dla danej klasy może być przesyłany jako ruch należący do niższej klasy lub po prostu odrzucony.



Rys.4. Elementy funkcjonalne w ruterach DiffServ. KL – klasyfikacja, PZ – pomiar zgodności, U – usuwanie, M – markowanie, MP – multipleksacja, K – kolejka, SP – szeregowanie pakietów

W architekturze *DiffServ* zasady PHB mogą być zaimplementowane za pomocą algorytmu kolejowania i zarządzania kolejkami. Ponadto można wyróżnić elementy funkcjonalne implementowane w ruterach, które mogą być stosowane w zależności od umiejscowienia rutera w sieci (brzegowy lub szkieletowy) oraz przyjętych reguł dla obsługi poszczególnych klas ruchu. Na rys.4 przedstawiono różne możliwości stosowania mechanizmów w ruterach *DiffServ* [3, 5]. Wyróżnia się następujące elementy funkcjonalne.

- Klasyfikator (*Classifier*) typu **BA** (*Behaviour Aggregate*) lub **MF** (*Multi-Field*); klasyfikuje pakiety IP w przypadku BA tylko na podstawie pola DSCP, natomiast w przypadku MF dodatkowo uwzględnia się inne informacje zawarte w nagłówku pakietu IP, jak np. adres źródłowy, numer portu itd.

- Urządzenie monitorujące (*Meter*) mierzy zgodność strumienia danych z parametrami zawartymi w kontrakcie **SLA** (*Service Level Agreement*) oraz umożliwia zbieranie statystyk ruchowych (najczęściej stosuje się algorytm typu *Token Bucket*).
- Marker znakuje lub usuwa pakiety niezgodne z parametrami zawartymi w kontrakcie ruchowym.
- Multiplekser multipleksuje strumienie danych.
- Urządzenie usuwające pakiety wyłącznie usuwa pakiety niezgodne z kontraktem ruchowym (np. algorytm **RED** – *Random Early Detection*).
- Kolejka do przechowywania pakietów w buforze (najczęściej stosuje się dyscyplinę obsługi typu **FIFO**).
- Urządzenie szeregujące pakiety realizuje algorytm szeregujący pakiety do obsługi z poszczególnych kolejek, najczęściej spotykane rozwiązania są oparte na priorytetach lub algorytmie **WFQ** (*Weighted Fair Queuing*).

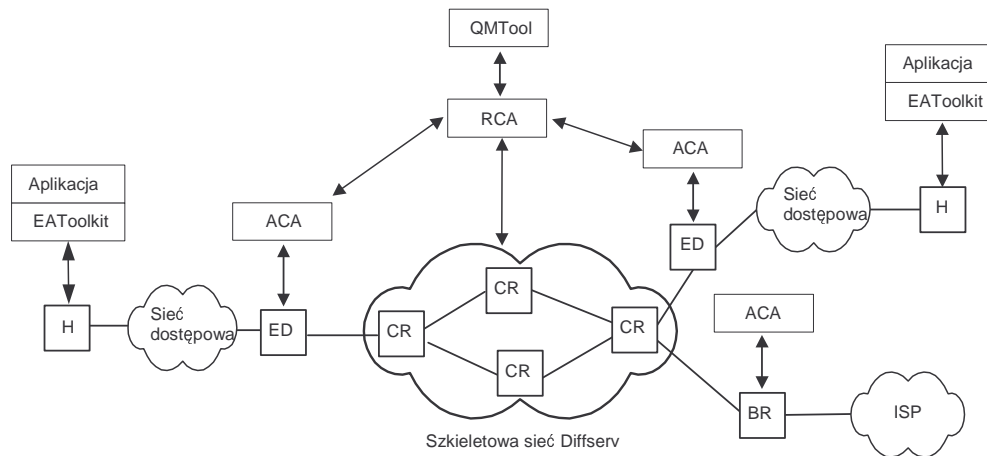
Oprócz wyżej wymienionych mechanizmów niezbędnym elementem funkcjonalnym w sieciach DiffServ jest tzw. broker pasma (*Bandwidth Broker*), realizujący funkcje zarządzania zasobami i decydujący o przyjmowaniu nowych strumieni danych do obsługi. Broker w ramach jednej domeny realizuje zasady dostępu poszczególnych użytkowników do wspólnych zasobów. Pomiędzy operatorem sieci DS a grupą użytkowników jest zawierany kontrakt, tzw. SLA. Obejmuje on specyfikację klas usług wraz z parametrami opisującymi dopuszczalny ruch, jaki użytkownicy mogą generować w ramach każdej z klas. Ponadto zawiera on informacje dotyczące adresów źródłowych i docelowych, numery portów, identyfikatory protokołu, aplikacji itd. Kontrakt ten może być zawarty w sposób statyczny (na dłuższy okres czasu) lub dynamiczny (w tym przypadku, konieczne jest użycie protokołu sygnalizacyjnego, np. RSVP).

3 ARCHITEKTURA I USŁUGI W SIECI AQUILA

Architektura sieci AQUILA [9, 10, 11, 12] jest oparta na architekturze DiffServ. Zakłada ona wyróżnienie dwóch typów elementów sieciowych, tj. urządzeń brzegowych ED (*Edge Devices*) i ruterów szkieletowych **CR** (*Core Routers*). Urządzenia brzegowe służą do podłączania do sieci użytkowników końcowych i implementują pełny zestaw mechanizmów związanych z obsługą pojedynczych strumieni ruchu; przykładowo klasyfikatorów czy też urządzeń monitorujących zgodność z deklarowanym profilem ruchowym wraz z markowaniem pakietów. Zgodnie z architekturą DiffServ, rutery szkieletowe nie rozróżniają

pojedynczych podstrumieni ruchu (*microflow*), operując jedynie na niewielkiej liczbie, w tym przypadku pięciu, zdefiniowanych klasach ruchu. Klasa ruchu obejmuje zagregowany strumień pakietów (*macroflow*), łącząc w sobie wiele podstrumieni. Pakiety przynależne danej klasie ruchu podlegają takim samym regułom obsługi w węźle sieci (PHB).

W rozważanej architekturze AQUILA (rys.5) zakłada się uzupełnienie architektury DiffServ dodatkową warstwą sterowania zasobami **RCL** (*Resource Control Layer*).



Rys.5. Ogólna architektura sieci AQUILA (RCA, ACA i EAToolkit tworzą warstwę RCL).

RCA – agent zarządzania zasobami, ACA – agent sterowania przyjmowaniem nowych wywołań, EAToolkit – interfejs do komunikacji pomiędzy aplikacją i ACA, H – terminal użytkownika, ED – urządzenie brzegowe, CR – ruter szkieletowy, BR – ruter brzegowy

Mechanizmy zaimplementowane w warstwie RCL odpowiadają za sterowanie i zarządzanie zasobami sieci; elementy warstwy RCL można więc identyfikować z płaszczyzną sterowania, podczas gdy sieć DiffServ realizuje funkcje płaszczyzny przekazu danych użytkowych. Dodatkowo warstwa RCL została podzielona na dwie podwarstwy, których funkcje są zaimplementowane w następujących agentach: **RCA** (*Resource Control Agent*) i **ACA** (*Admission Control Agent*). Agent ACA odpowiada za sterowanie przyjmowaniem nowych wywołań do sieci, podczas gdy agent RCA realizuje funkcje związane z zarządzaniem zasobami w sieci, w tym dystrybucją/redystrybucją zasobów transmisyjnych dla poszczególnych urządzeń brzegowych, skojarzonych z agentami ACA.

Zakłada się, że użytkownik korzysta z aplikacji wyposażonej w specjalny interfejs dla przesłania do ACA deklaracji odnośnie generowanego ruchu i swoich żądań dotyczących jakości obsługi (np. przez podanie odpowiedniej usługi sieciowej).

W sieci AQUILA zdefiniowano zestaw czterech usług sieciowych **NS** (*Network Services*), które zapewniają użytkownikom zróżnicowany poziom gwarancji jakości obsługi. Usługi

sieciowe różnią się pod względem zapewnianych wartości parametrów QoS, typem obsługiwanego ruchu (strumieniowy czy elastyczny) i sposobem jego charakteryzacji. Każda usługa sieciowa jest zaprojektowana dla efektywnego przekazu ruchu wysyłanego przez aplikacje charakteryzujące się podobnymi wymaganiami QoS. Poszczególne usługi sieciowe są mapowane na tzw. klasy ruchu **TCL** (*Traffic Class*), w ramach których definiuje się mechanizmy obsługi ruchu w urządzeniach brzegowych, zasady przyjmowania nowych wywołań i mechanizmy PHB.

3.1 Usługi w sieci AQUILA

W architekturze AQUILA zdefiniowano cztery usługi QoS i jedną usługę bez QoS. Usługi QoS są następujące [9]: Premium CBR (**PCBR**), Premium VBR (**PVBR**), Premium Multimedia (**PMM**) oraz Premium Mission Critical (**PMC**). Dwie pierwsze usługi są przeznaczone dla obsługi ruchu strumieniowego, a dwie następne dla ruchu elastycznego. Usługa Standard (**STD**) odpowiada przekazowi pakietów na zasadzie *best effort*.

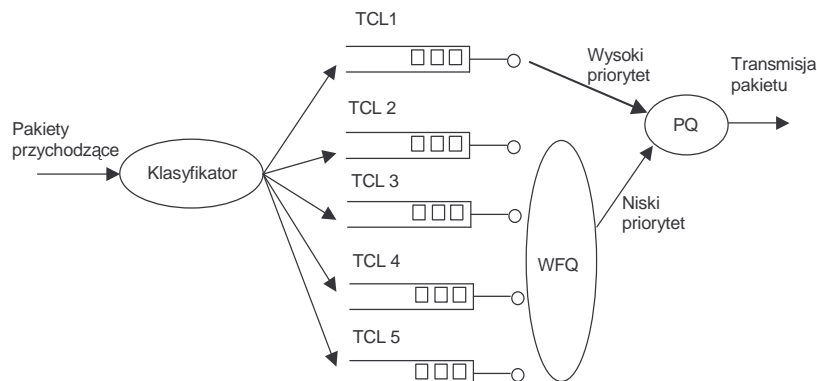
Usługę PCBR zaprojektowano dla przekazu ruchu o stałej szybkości bitowej ze ścisłymi gwarancjami zapewnienia małego opóźnienia przekazu pakietów i małego poziomu utraty pakietów. Ruch o takim profilu i takich wymaganiach QoS jest właściwy dla aplikacji opartych na przekazie mowy. Ruch kierowany do sieci przez użytkownika jest w tym przypadku scharakteryzowany jedynie przez podanie szczytowej szybkości bitowej **PBR** (*Peak Bit Rate*). Usługę PVBR przeznaczono dla ruchu o zmiennej szybkości bitowej. Gwarancje QoS zapewniane przez tę usługę są podobne do oferowanych w przypadku usługi PCBR. Typowym zastosowaniem usługi PVBR jest przesyłanie obrazów wideo. Ruch użytkownika jest w tym przypadku charakteryzowany przez podanie PBR i granicznej średniej szybkości bitowej **SBR** (*Sustainable Bit Rate*).

Pozostałe usługi sieciowe QoS są przeznaczone do przekazu ruchu elastycznego, tj. wykorzystującego protokół **TCP** (*Transmission Control Protocol*). Ruch kierowany do usługi PMM jest charakteryzowany tylko przez podanie wartości SBR. Rozważaną usługę przeznaczono głównie do przesyłania ruchu związanego z długotrwałymi połączeniami TCP, w których źródło ma charakter „zachłanny” (*greedy source*). Z drugiej strony usługa PMC jest przeznaczona do przesyłania ruchu generowanego przez krótkotrwałe połączenia TCP. W tym przypadku ruch jest scharakteryzowany, podobnie jak w usłudze PVBR, przez podanie wartości PBR i SBR.

3.2 Mechanizmy obsługi ruchu na poziomie pakietów

Ruch kierowany do usług sieciowych jest obsługiwany zgodnie z zasadami związanymi z odpowiednimi klasami ruchu TCL [9]. Zależnie od klasy TCL, monitorowanie zgodności ruchu użytkownika z profilem ruchowym jest realizowane poprzez pojedynczy lub podwójny algorytm **TB** (*Token Bucket*). Klasa **TCL1** (związana z usługą PCBR) wykorzystuje pojedynczy algorytm TB, podczas gdy klasa **TCL2** (związana z usługą PVBR) podwójny algorytm TB. W obu przypadkach ruch nadmiarowy (tj. niezgodny z profilem ruchowym określonym podczas fazy ustanawiania rezerwacji zasobów) jest usuwany z sieci. W przypadku klasy **TCL3** (związanej z usługą PMM) zastosowano pojedynczy algorytm TB, podczas gdy klasa **TCL4** (związana z usługą PMC) wykorzystuje podwójny algorytm TB. W tych klasach ruch nadmiarowy jest oznaczany jako niezgodny z profilem (*out-of-profile*) i jest on dopuszczony do sieci, a więc inaczej traktowany niż w klasach TCL1 i TCL2.

Po przejściu ruchu przez mechanizmy klasyfikacji, monitorowania i markowania (co jest realizowane jedynie w urządzeniach brzegowych, a nie w ruterach szkieletowych) pakiety są następnie buforowane w odpowiednich kolejkach wyjściowych, w zależności od przynależności do klasy TCL. Schemat kolejkowania przedstawiono na rys.6. Kolejki związane z klasami TCL1 i TCL2 wykorzystują prosty mechanizm usuwania pakietów z końca kolejki (*drop tail*). Klasy TCL3 i TCL4, które obsługują ruch o charakterze elastycznym, wykorzystują algorytm zarządzania buforem **WRED** (*Weighted Random Early Discard*), który pozwala na zróżnicowanie dostępu do bufora dla pakietów zgodnych i niezgodnych z profilem ruchowym. Pakiety należące do klasy TCL1 są obsługiwane z wyższym priorytetem (typu *non-preemptive*), natomiast pozostałe kolejki zgodnie z algorytmem szeregowania pakietów WFQ. Wykorzystany w architekturze AQUILA algorytm szeregowania pakietów w porcie wyjściowym rutera jest równoważny mechanizmowi **CBWFQ** (*Class Based Weighted Fair Queueing*), zaimplementowanemu w ruterach firmy **CISCO**.

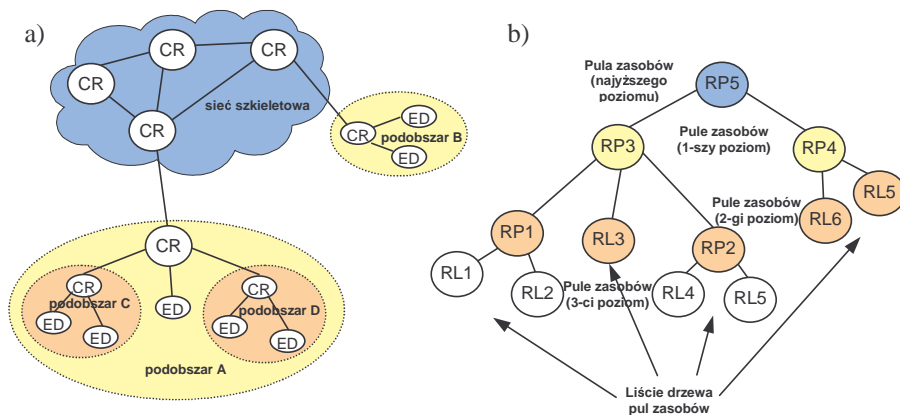


Rys.6. Architektura portu wyjściowego rutera. TCL – klasa ruchu, WFQ – algorytm Weighted Fair Queueing, PQ – algorytm Priority Queueing

3.3 Zarządzanie zasobami

Mechanizmy dystrybucji/redystrybucji zasobów działają oddzielnie dla każdej klasy TCL. Oznacza to, iż zasoby przypisane dla danej klasy nie mogą być wykorzystywane przez inne klasy. W konsekwencji zasoby sieci są w zasadzie podzielone na 5 podsieci, związanych z zasobami przypisanymi dla poszczególnych klas TCL. Z każdym urządzeniem ED jest związany agent ACA. Realizuje on wywołania przychodzące od użytkowników podłączonych do danego urządzenia ED do limitu przydzielonych zasobów (*AC limit*). Limit ten jest obliczany na podstawie przewidywanego zapotrzebowania ruchowego pomiędzy relacjami ED-ED, routingu w sieci itp. Limit zasobów przypisany dla danego ACA reprezentuje sumaryczny ruch, jaki może być przesłany do sieci przez dany ED, bez względu na docelowy ED.

Precyzyjne przewidywanie zapotrzebowania ruchowego w sieci Internet jest raczej trudne. Dlatego też w sieci AQUILA zastosowano adaptacyjny mechanizm zarządzania zasobami, który zapewnia zwiększenie stopnia wykorzystania sieci. Mechanizm ten realizuje „pożyczanie” zasobów pomiędzy agentami ACA w przypadku, gdy np. wykorzystanie zasobów w jednym z ACA jest mniejsze od zadanej wartości progowej. Zasoby przydzielone dla poszczególnych agentów ACA są pogrupowane w tzw. pule zasobów (*Resource Pools*) [11]. Reprezentują one część zasobów sieci, które mogą być wspólnie wykorzystywane przez pewną liczbę agentów ACA. Pule zasobów mogą być dalej grupowane, tworząc pule zasobów wyższego poziomu. Hierarchie pul zasobów mogą być w łatwy sposób tworzone dla sieci o topologii drzewiastej. W najprostszym przypadku pula zasobów może odpowiadać węzłowi w sieci o topologii drzewa, przy czym liście w tym drzewie są też liśćmi w drzewie pul zasobów (odpowiadają im poszczególne ACA). To podejście może być rozszerzone na podobszary sieci, które charakteryzują się strukturą drzewiastą.



Rys.7. a) Przykładowa struktura sieci IP i b) odpowiadająca jej hierarchia pul zasobów. CR – ruter szkieletowy, ED – urządzenie brzegowe, RP – pula zasobów, RL – liść w drzewie pul zasobów

Na rys.7 przedstawiono przykładową hierarchię pul zasobów. Korzeń drzewa reprezentuje całkowite zasoby transmisyjne sieci szkieletowej. W rozważanej strukturze sieci można wyróżnić dwa podobszary (A i B), dla których utworzono pule zasobów RP3 i RP4. Pule RP3 i RP4 tworzą pierwszy stopień w hierarchii pul zasobów. Zasoby zgrupowane w RP3 i RP4 są dalej dzielone pomiędzy pule zasobów niższego poziomu. Ten proces jest powtarzany aż do najniższego stopnia w hierarchii pul zasobów, reprezentowanego przez liście w drzewie (odpowiadające poszczególnym ACA).

3.4 Sterowanie przyjmowaniem nowych wywołań

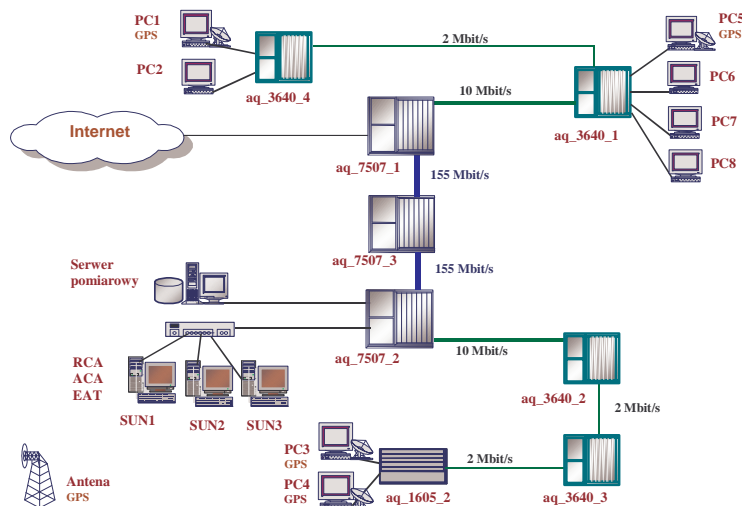
Standardowa sieć IP oferuje użytkownikom tylko jedną usługę, opartą na przekazie pakietów na zasadzie *best-effort*. Ta usługa nie gwarantuje jakości obsługi, więc sterowanie przyjmowaniem zgłoszeń do sieci nie jest konieczne. W konsekwencji brak jest mechanizmów zapewniających kontrolę nad liczbą strumieni pakietów obsługiwanych w sieci w tym samym czasie. W projekcie AQUILA zdefiniowano zestaw nowych usług, które zapewniają różne gwarancje QoS. Dlatego też zaistniała konieczność wprowadzenia metod sterowania przyjmowaniem nowych wywołań **AC** (*Admission Control*), które zapewniają spełnienie określonych wymagań dotyczących jakości obsługi. W sieci AQUILA odpowiednia jakość obsługi jest zapewniana poszczególnym podstrumieniom pakietów. Strumienie te są definiowane jako jednokierunkowe przepływy pakietów, które mogą być rozróżniane na wejściu do sieci przez rozpoznawanie wartości pól nagłówka pakietu IP (adres źródłowy i docelowy, typ protokołu itp.). Taka definicja strumienia pakietów jest kompatybilna z terminologią używaną w specyfikacji architektury DiffServ.

Ogólnie, wyróżnia się dwa typy metod sterowania przyjmowaniem nowych wywołań: algorytmy biorące pod uwagę tylko parametry ruchowe zadeklarowane przez użytkownika oraz algorytmy dodatkowo wspomagane pomiarami ruchu w sieci. W pierwszej fazie projektu AQUILA zaimplementowano i przebadano metody należące do pierwszej grupy. Zgłoszenia użytkowników (przesyłane do sieci za pomocą specjalnie opracowanej procedury sygnalizacyjnej) są przyjmowane lub odrzucane na podstawie wartości parametrów deskryptora ruchowego, deklarowanych przez użytkownika. Celem badań przeprowadzonych w ramach projektu AQUILA było sprawdzenie, czy metody AC opracowane dla zastosowania w sieci ATM [7, 8] mogą być efektywnie zaadaptowane do sieci IP QoS.

Zgodnie z architekturą sieci AQUILA, procedura podejmowania decyzji o przyjęciu/odrzuceniu zgłoszenia jest realizowana w sieci tylko w punkcie wejściowym (*ingress*) i wyjściowym (*egress*). Brak możliwości sprawdzenia AC na każdym kolejnym łączy w sieci sprawia, że rzeczywista znajomość dostępności wolnych zasobów w sieci jest obciążona pewną niepewnością. Znacznie utrudnia to sterowanie przyjmowaniem nowych wywołań. Dla przyjmowania nowych wywołań na wejściu do sieci założono tzw. model pojedynczego łącza. Oznacza to, że algorytmy AC są stosowane tak, jakby zasoby transmisyjne sieci były reprezentowane jako jedno łącze charakteryzujące się pewną przepływnością (**C**) i pojemnością bufora (**B**). Wartość parametru C odpowiada wielkości zasobów zaalokowanych dla danego ACA, natomiast pojemność bufora odpowiada zdolności buforowania sieci (obecnie jest ona reprezentowana przez rozmiar najmniejszego bufora we wszystkich ruterach w sieci). Istotne jest to, że każda klasa ruchu (TCL) jest reprezentowana poprzez oddzielne wartości parametrów C i B, odpowiadające pojemności łącza i bufora zarezerwowanej dla tej klasy ruchu. Szczegóły dotyczące algorytmów AC zostały opisane w [12].

3.5 Wyniki badań sieci pilotowej projektu AQUILA

Działanie sieci zaimplementowanej zgodnie z architekturą AQUILA zostało częściowo sprawdzone eksperymentalnie w sieci pilotowej zainstalowanej w Centrum Badawczo-Rozwojowym TP SA w Warszawie. Topologię sieci pilotowej przedstawiona na rys. 8 [10]. Składa się ona z 8 ruterów, połączonych w konfiguracji łańcucha. Sieć pilotową oparto na urządzeniach firmy CISCO typu 7507, 3640 i 1605, wyposażonych w odpowiednie mechanizmy QoS.



Rys.8. Topologia sieci pilotowej IP QoS AQUILA. PC – terminal użytkownika (lub generator/analizator ruchu)

3.5.1 Badanie przekazu mowy w sieci AQUILA

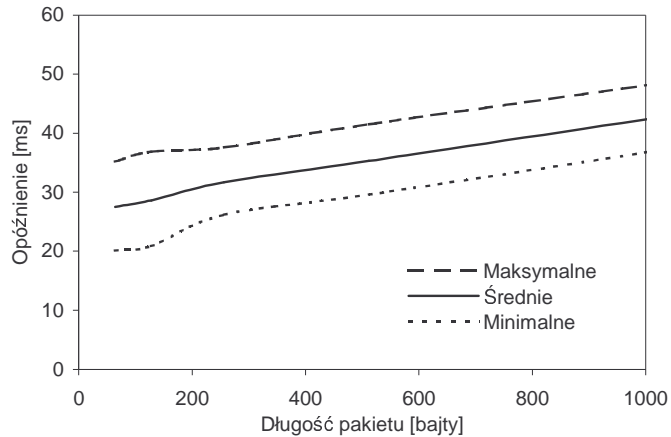
Możliwości przesyłania ruchu typu audio przez sieć IP z jakością obsługi zostały sprawdzone na przykładzie eksperymentalnej aplikacji telefonii internetowej **WinSIP** firmy **Siemens**. Ruch generowany przez tę aplikację charakteryzuje się stałą szybkością bitową, wynoszącą 64 kbit/s (71,4 kbit/s po dodaniu nadmiaru związanego z nagłówkami protokołów UDP/IP/Ethernet) i stałą wielkością pakietu, wynoszącą 512 bajtów.

Ruch testowy był generowany w sieci pomiędzy dwoma terminalami (PC3 i PC1 na rys. 8), podłączonymi do ruterów brzegowych. Ruch testowy był więc przesyłany przez wszystkie 8 węzłów sieci pilotowej. Ruch podkładowy, którego celem było doprowadzenie w sieci testowej do warunków silnego przeciążenia, był generowany zgodnie ze wzorcem ruchu typu ON-OFF i wprowadzany do sieci na każdym kolejnym stopniu (na każdym łączu pomiędzy ruterami).

W pierwszym eksperymencie był generowany sztuczny ruch testowy o stałej szybkości bitowej wynoszącej 72 kbit/s i stałej długości pakietu. Charakterystyka takiego ruchu odpowiada profilowi ruchu generowanego przez aplikację WinSIP. W trakcie eksperymentu zmierzono opóźnienie przejścia przez sieć pakietów generowanych w ramach strumienia testowego. Pakiety przesyłane za pomocą usługi PCBR, przeznaczonej do przekazywania sygnału mowy, są transmitowane w sieci z najwyższym priorytetem. Opóźnienie, jakiego doznają te pakiety w sieci, jest więc głównie powodowane przez dwa czynniki: 1) opóźnienie transmisji pakietu TCL1, wynikające z tego, iż w momencie jego przyjscia w nadawczym buforze transmisyjnym są już pakiety o niższym priorytecie (mechanizm *TX Ring*) oraz 2)

czasu transmisji przez łącze. Wyniki pomiarów minimalnego, maksymalnego i średniego opóźnienia przejścia pakietów testowych przez sieć zostały przedstawione na rys.9.

Można przyjąć, że maksymalne dopuszczalne opóźnienie pakietów, przy którym jakość przesyłanego sygnału mowy jest akceptowalna dla użytkownika, wynosi 150 ms. Wykonane testy świadczą, że usługa PCBR w sieci IP QoS AQUILA spełnia to wymaganie.



Rys.9. Pomiar opóźnienia przekazu pakietów przez sieć

Wnioski z przedstawionych badań zostały potwierdzone w doświadczeniu z wykorzystaniem rzeczywistej aplikacji telefonii internetowej. Osoby wykonujące test przeprowadziły rozmowy przy użyciu aplikacji WinSIP. Warunki ruchowe w sieci były takie same jak w testach z użyciem sztucznego generatora ruchu. Można zatem wnioskować, iż jakość obsługi dla ruchu generowanego przez rzeczywistą aplikację jest podobna jak w przeprowadzonym eksperymencie, którego wyniki przedstawiono na rys.9 (dla przypomnienia w aplikacji WinSIP długość pakietu była stała i wynosiła 512 bajtów). Minimalne, maksymalne i średnie opóźnienie przejścia pakietu przez sieć wynosiło odpowiednio 29,6, 41,5 i 35,3 milisekundy. Osoby, które dokonywały subiektywnej oceny jakości przekazu mowy, uznały jakość działania aplikacji za akceptowalną. Nie zauważono różnicy pomiędzy odbieraną jakością przekazu mowy w sieci nieobciążonej i przeciążonej. Oczywiście podobny wynik nie byłby możliwy do uzyskania w sieci nie oferującej gwarancji jakości obsługi.

3.5.2 Przekaz obrazu wideo za pomocą aplikacji RealPlayer

Aplikacja **RealPlayer** służy do przesyłania w sieci i odtwarzania na komputerze użytkownika zakodowanego sygnału audio lub wideo (np. filmów, utworów muzycznych itp.) bez zachowania reżimu czasu rzeczywistego. Ze względu na brak tego reżimu, gwarancje dotyczące opóźnienia pakietów nie są w tym przypadku wymagane. Dlatego też jakość obsługi jest związana z gwarancją minimalnej przepływności dostępnej w sieci dla danej sesji

aplikacji. W sieci AQUILA jakość obsługi wymaganą przez aplikację RealPlayer oferuje usługa PMM.

W sieci testowej serwer **RealServer**, w którym były przechowywane pliki z fragmentami filmów wideo, zainstalowano na terminalu PC4. Klient aplikacji RealPlayer, który odbierał dane z serwera i odtwarzał film na komputerze użytkownika, był zainstalowany w terminalu PC2. Rozważano dwa przypadki testowe. W przypadku #1, który odpowiada sytuacji obserwowanej w obecnej sieci Internet, ruch związany z aplikacją RealPlayer był obsługiwany na zasadzie *best-effort*. Łącze o szybkości 2 Mbit/s pomiędzy ruterem ED a siecią szkieletową zostało obciążone ruchem podkładowym generowanym ze stałą szybkością bitową, równą 1750 kbit/s i 2200 kbit/s.

W przypadku testowym #2 ruch związany z aplikacją RealPlayer był przesyłany w sieci AQUILA przy użyciu usługi PMM, z deklaracją wartości SBR na poziomie 248 kbit/s i 264 kbit/s. Dodatkowo ruch generowany do usługi PMM był uzupełniony przez 4 „zachłanne” źródła TCP, które emulowały przekaz plików za pomocą protokołu FTP. Sumaryczna wartość deklarowanych SBR przez użytkowników nadających w ramach usługi PMM wynosiła 600 kbit/s, czyli była równa pojemności dedykowanej dla tej usługi na łączu pomiędzy urządzeniem ED a siecią szkieletową. W pozostałych usługach sieciowych był generowany sztuczny ruch podkładowy wypełniający przydzielone im wartości *AC limit*. Całkowity ruch oferowany na łącze do sieci szkieletowej wynosił 2450 kbit/s, a więc można uznać, iż łącze wychodzące z urządzenia ED było w stanie przeciążenia. Wyniki eksperymentów przedstawiono w tabeli 2 i w tabeli 3.

Tabela 2. Przypadek #1: Aplikacja RealPlayer działająca w standardowej sieci IP. Warunki ruchowe zmieniają się od stanu nasycenia do stanu przeciążenia.

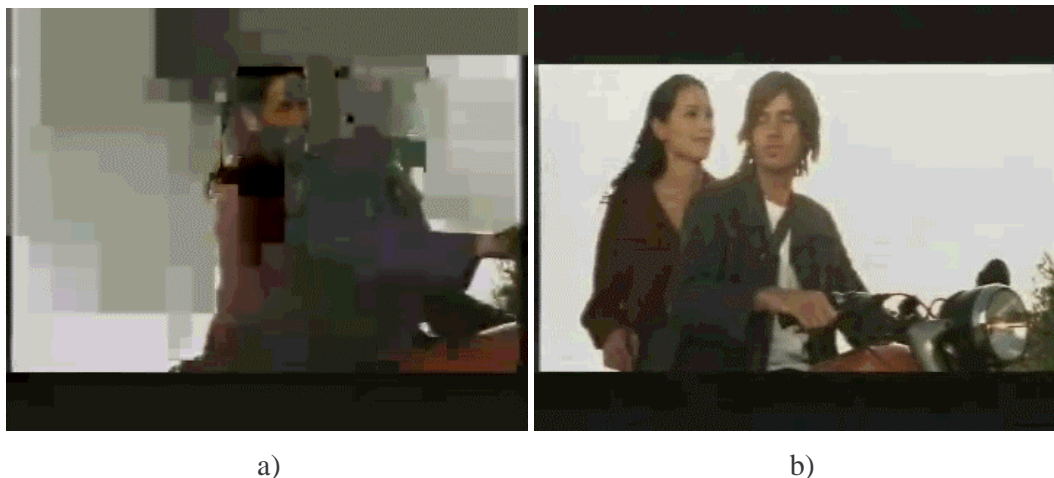
Ruch podkładowy	Całkowity ruch generowany do sieci	Subiektywna ocena jakości działania aplikacji RealPlayer
1750 kbit/s	2000 kbit/s	jakość dobra, brak zakłóceń dźwięku i obrazu
2200 kbit/s	2450 kbit/s	jakość zła, nieakceptowalne zakłócenia dźwięku i obrazu

Tabela 3. Przypadek #2: Aplikacja RealPlayer działająca w sieci IP QoS z wykorzystaniem usługi PMM. Łącze do sieci szkieletowej pracuje w warunkach przeciążenia.

Wartości parametrów SR dla pięciu rezerwacji w usłudze PMM	Całkowity ruch generowany do sieci	Subiektywna ocena jakości działania aplikacji RealPlayer
SBR = 248 kbit/s dla Real Player	2450 kbit/s	jakość niezadowolająca, zaobserwowano pewne zakłócenia dźwięku i obrazu przeszkadzające w

SBR = 88 kbit/s dla 4 połączeń TCP		odbiorze przekazu
SBR = 264 kbit/s dla Real Player SBR = 80 kbit/s dla 4 połączeń TCP	2450 kbit/s	jakość dobra, brak zakłóceń dźwięku i obrazu

Jak można było oczekiwać, jakość działania aplikacji w sieci w stanie przeciążenia i bez mechanizmów zapewniających jakość obsługi (przypadek #1) była nie do zaakceptowania. W sieci IP z zastosowanymi tego rodzaju mechanizmami (przypadek #2) ruch generowany przez aplikację RealPlayer był przesyłany z zagwarantowaniem odpowiednich parametrów QoS (w tym przypadku minimalnej gwarantowanej przepływności SBR). Jakość działania aplikacji była taka sama jak w sieci nie obciążonej. Warunkiem uzyskania tego wyniku była prawidłowa deklaracja parametrów ruchowych. Porównanie obrazu otrzymanego na ekranie komputera w wyniku działania aplikacji RealPlayer w standardowej sieci IP w stanie przeciążenia i w sieci z mechanizmami zapewniającymi jakość obsługi przedstawiono na rys. 10.



Rys.10. Ekran aplikacji RealPlayer uruchomionej w warunkach przeciążenia a) w sieci tradycyjnej i b) w sieci z jakością obsługi, z parametrem SBR = 264 kbit/s

4 PODSUMOWANIE

W artykule przedstawiono wyniki eksperymentów dowodzących, iż sieć IP z mechanizmami gwarantującymi jakość przekazu może spełniać wymagania stawiane sieciom wielousługowym. Przeprowadzone eksperymenty dotyczyły jakości przekazu ruchu generowanego przez aplikacje oparte na przekazie mowy i wideo, a więc aplikacje wprawdzie dostępne w sieci Internet, lecz jednak mało popularne ze względu na obecną jej jakość.

Opisana w artykule architektura AQUILA, oparta na architekturze DiffServ, została zaimplementowana w postaci laboratoryjnej sieci testowej w **Centrum Badawczo-Rozwojowym (CBR) TP SA**. Architektura ta została opracowana w ramach Projektu Europejskiego AQUILA, w którym ze strony polskiej bierze aktywny udział **Instytut Telekomunikacji Politechniki Warszawskiej** i CBR TP SA. Przedstawione wyniki należy traktować jako uzyskane w sieci eksperymentalnej, opartej na rozwiązaniach prototypowych. Należy jednak oczekiwać, iż w najbliższych latach będzie się dążyć do osiągnięcia przez sieć Internet funkcjonalności, jaka cechuje sieć AQUILA.

5 LITERATURA

- [1] Braden R., Clark D. and Shenker S.: *Integrated Services in the Internet Architecture: an Overview*, RFC 1633, czerwiec 1994
- [2] Braden R. i inni: *Resource ReSerVation Protocol (RSVP) — Version 1 Functional Specification*, RFC 2205, wrzesień 1997
- [3] Blake S. i inni: *An Architecture for Differentiated Services*, RFC 2475, grudzień 1998
- [4] Bernet Y. i inni: *Differentiated Services Framework*, Internet Draft, draft-ietf-diffserv-framework-0.2.txt, Luty 1999
- [5] Bernet Y. i inni: *A Conceptual Model for DiffServ Routers*, internet-draft, draft-ietf-diffserv-model-02.txt, Wrzesień 2000
- [6] Nichols K. i inni: *Definition of the Differentiated Services Field (DS Field) in the IPv4 and IPv6 Headers*, RFC 2474, grudzień 1998
- [7] Roberts J., Mocchi U., Virtamo J.: Raport końcowy projektu COST 242, *Broadband network teletraffic: Performance evaluation and design of broadband multiservice networks*, Lectures Notes in Computer Science 1155, Springer 1996
- [8] Tran-Gia P., Vicari N.: Raport końcowy projektu COST 257, *Impact of New Services on the Architecture and Performance of Broadband Networks*, compuTEAM, Wuerzburg, Niemcy 2000
- [9] Konsorcjum projektu AQUILA: *Deliverable D1301, Specification of traffic handling for the first trial*, <http://www.ist-aquila.org>, wrzesień 2000
- [10] Konsorcjum projektu AQUILA: *Deliverable D3201, First Trial Report*, <http://www.ist-aquila.org>, kwiecień 2001

- [11] Bak A., Burakowski W, Ricciato F., Salsano S., Tarasiuk H.: *Traffic handling in AQUILA QoS IP network*, 2nd International Workshop on Quality of future Internet Services, 24-26 wrzesień 2001, Coimbra, Portugal
- [12] Burakowski W. i inni: *AQUILA network architecture: first trial experiments*, ukaże się w Special Issue of Journal of Telecommunications and Information Technology, Warszawa 2002
- [13] Burakowski W. i inni, *Sieci IP z jakością obsługi*, Konferencja Internet 2000, Wrocław, 2000

Tabela 1: Wymagania dotyczące jakości przekazu dla wybranych usług multimedialnych

Usługi multimedialne	Wrażliwość na:		Przykłady
	straty pakietów	opóźnienia	
Interaktywne wideo: czasu rzeczywistego małej rozdzielczości	duża duża	duża średnia	wideokonferencja <i>CU-SeeMe</i> (PC komputer)
Odtwarzanie wideo	duża	mała	wideo na żądanie wideo ze strony Web
Stały obraz	duża	mała	obraz ze strony Web
Faks	mała	mała	zdalne kopiowanie
Interaktywna mowa Odtwarzanie mowy	mała mała	duża mała	telefonía mowa ze strony Web
Interaktywne dane: dużej szybkości małej szybkości Nie-interaktywne dane	duża duża duża	duża średnia mała	sterowanie w czasie rzeczywistym Telnet E-mail

Tabela 2. Przypadek #1: Aplikacja RealPlayer działająca w standardowej sieci IP. Warunki ruchowe zmieniają się od stanu nasycenia do stanu przeciążenia.

Ruch podkładowy	Całkowity ruch generowany do sieci	Subiektywna ocena jakości działania aplikacji RealPlayer
1750 kbit/s	2000 kbit/s	jakość dobra, brak zakłóceń dźwięku i obrazu
2200 kbit/s	2450 kbit/s	jakość zła, nieakceptowalne zakłócenia dźwięku i obrazu

Tabela 3. Przypadek #2: Aplikacja RealPlayer działająca w sieci IP QoS z wykorzystaniem usługi PMM. Łącze do sieci szkieletowej pracuje w warunkach przeciążenia.

Wartości parametrów SR dla pięciu rezerwacji w usłudze PMM	Całkowity ruch generowany do sieci	Subiektywna ocena jakości działania aplikacji RealPlayer
SBR = 248 kbit/s dla Real Player SBR = 88 kbit/s dla 4 połączeń TCP	2450 kbit/s	jakość niezadowolająca, zaobserwowano pewne zakłócenia dźwięku i obrazu przeszkadzające w odbiorze przekazu
SBR = 264 kbit/s dla Real Player SBR = 80 kbit/s dla 4 połączeń TCP	2450 kbit/s	jakość dobra, brak zakłóceń dźwięku i obrazu

Streszczenie

Przedstawiono wymagania stawiane sieci wielousługowej, tj. realizującej przekaz informacji związanej z aplikacjami opartymi na mowie, danych i wideo. Po krótkim podsumowaniu technik N-ISDN i ATM, uwagę skupiono na sieci IP (Internet), sieci IP QoS nowej generacji, tj. z zaimplementowanymi mechanizmami QoS. W szczególności została przedstawiona architektura AQUILA (opracowana w ramach 5. Programu Ramowego Unii Europejskiej) oraz rezultaty eksperymentów przeprowadzonych w instalacji pilotowej w Centrum Badawczo-Rozwojowym TP SA. Uzyskane wyniki wskazują jednoznacznie, iż sieć IP QoS spełnia wymagania stawiane sieciom wielousługowym.

Słowa kluczowe: IP QoS, sieć wielousługowa, pomiary w sieci pilotowej

Abstract

Multi-service IP QoS network: architecture and practical experiments

The paper summarizes the requirements for multi-service networks, with their capabilities for effective transfer of a variety of applications based on voice, computer data and video transfer. After short overview of N-ISDN and ATM, the main focus is put on the new generation of IP-based network (Internet), called IP QoS. More precisely, an outline of AQUILA architecture (developed by the 5. Framework European Project) was done as well as the results of practical experiments using testbed installed in the Research and Development Center of Telekomunikacja Polska S.A. were described. The obtained results say that IP QoS network concept confirm the expectations for multi-service network.

Keywords: IP QoS, multi-service network, testbed measurements

Podpisy pod rysunkami

Rys.1. Przepływ wiadomości sygnalizacyjnych w protokole RSVP.

Rys.2. Ogólna architektura DiffServ

Rys.3. Format oktetu IPv4 TOS i pola DS (Differentiated Services).

Rys.4. Elementy funkcjonalne w ruterach DiffServ. KL – klasyfikacja, PZ – pomiar zgodności, U – usuwanie, M – markowanie, MP – multipleksacja, K – kolejka, SP – szeregowanie pakietów

Rys.5. Ogólna architektura sieci AQUILA (RCA, ACA i EAToolkit tworzą warstwę RCL).

RCA – agent zarządzania zasobami, ACA – agent sterowania przyjmowaniem nowych wywołań, EAToolkit – interfejs do komunikacji pomiędzy aplikacją i ACA, H – terminal użytkownika, ED – urządzenie brzegowe, CR – ruter szkieletowy, BR – ruter brzegowy

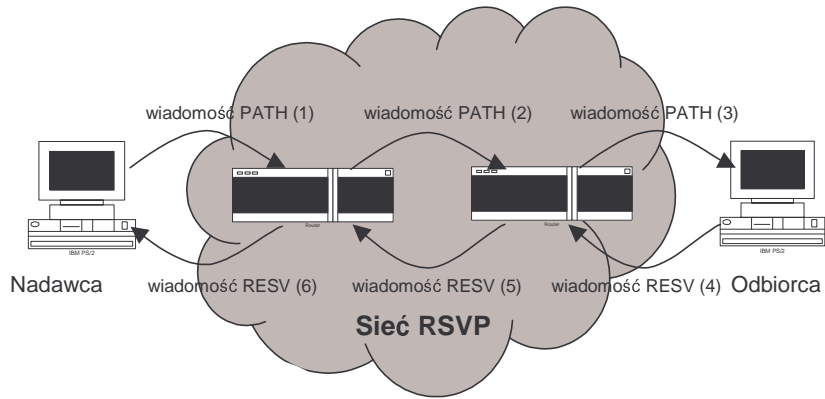
Rys.6. Architektura portu wyjściowego rutera. TCL – klasa ruchu, WFQ – algorytm Weighted Fair Queueing, PQ – algorytm Priority Queueing

Rys.7. a) Przykładowa struktura sieci IP i b) odpowiadająca jej hierarchia pul zasobów. CR – ruter szkieletowy, ED – urządzenie brzegowe, RP – pula zasobów, RL – liść w drzewie pul zasobów

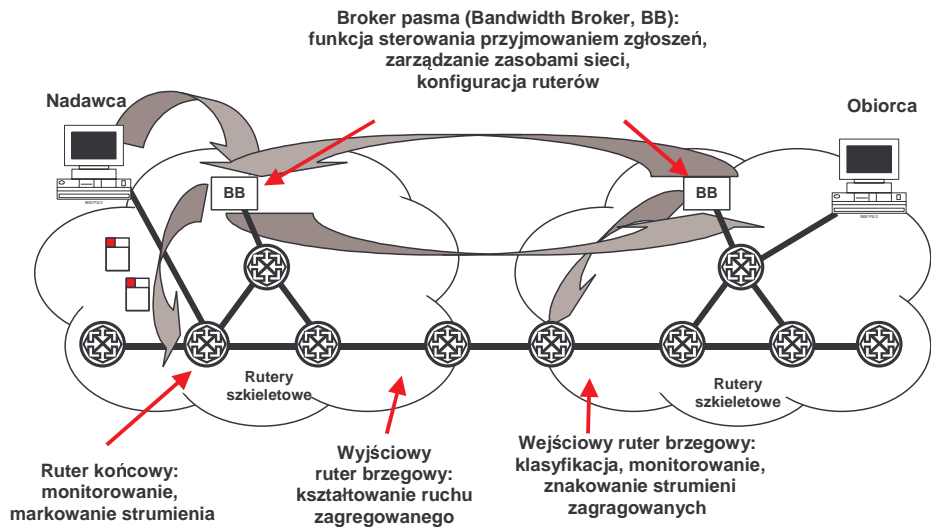
Rys.8. Topologia sieci pilotowej IP QoS AQUILA. PC – terminal użytkownika (lub generator/analizator ruchu)

Rys.9. Pomiar opóźnienia przekazu pakietów przez sieć

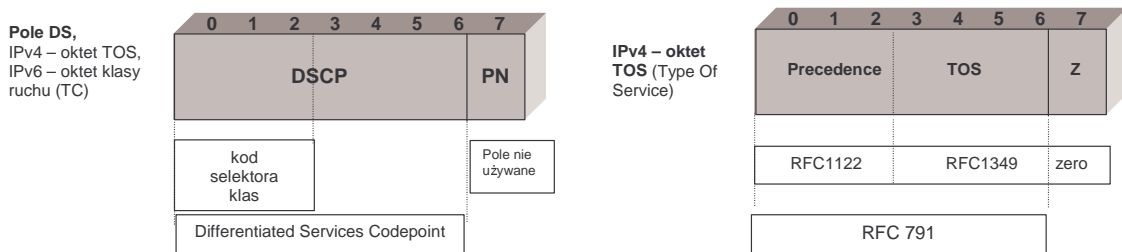
Rys.10. Ekran aplikacji RealPlayer uruchomionej w warunkach przeciążenia a) w sieci tradycyjnej i b) w sieci z jakością obsługi, z parametrem SBR = 264 kbit/s



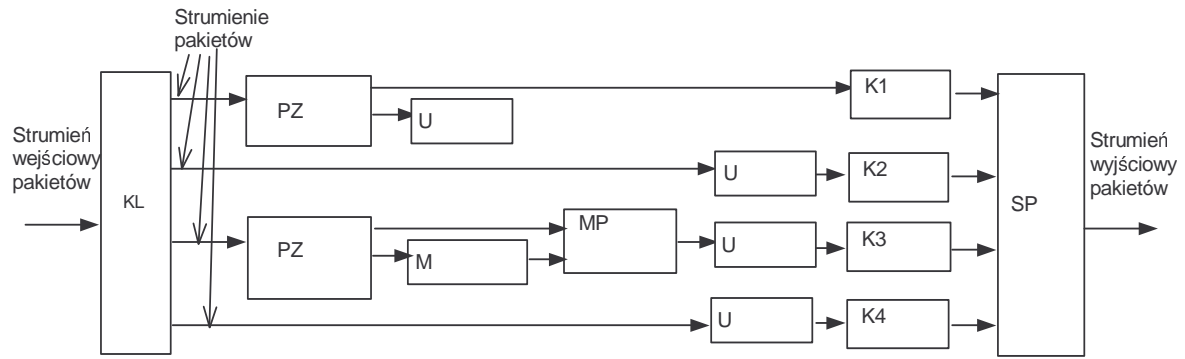
Rys.1. Przepływ wiadomości sygnalizacyjnych w protokole RSVP.



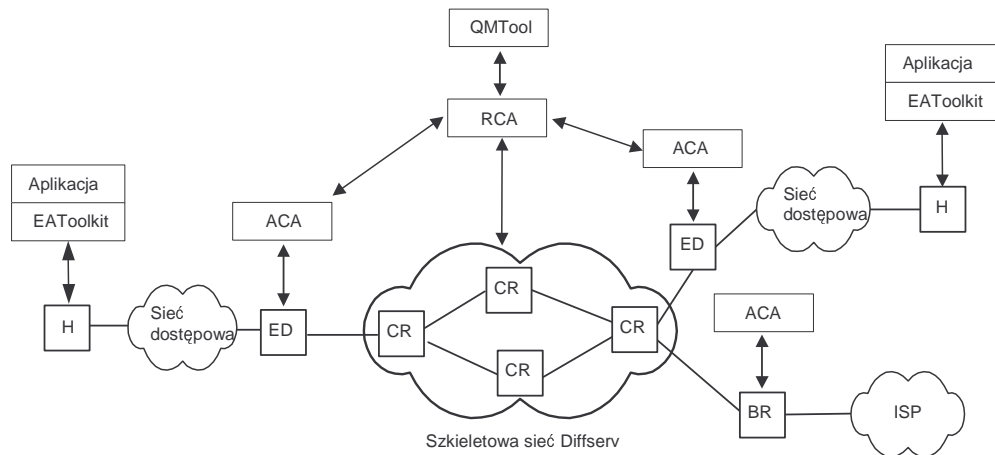
Rys.2. Ogólna architektura DiffServ.



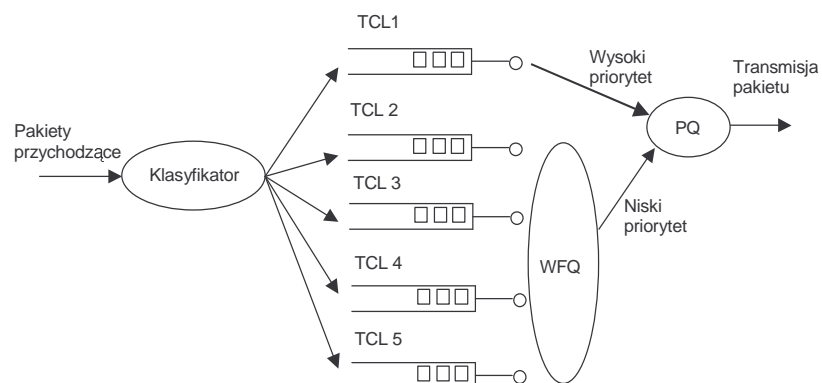
Rys.3. Format oktetu IPv4 TOS i pola DS (Differentiated Services).



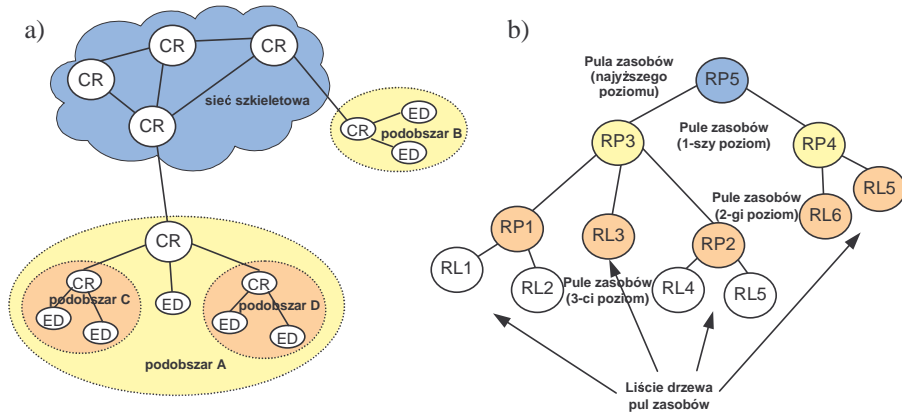
Rys.4. Elementy funkcjonalne w ruterach DiffServ. KL – klasyfikacja, PZ – pomiar zgodności, U – usuwanie, M – markowanie, MP – multipleksacja, K – kolejka, SP – szeregowanie pakietów



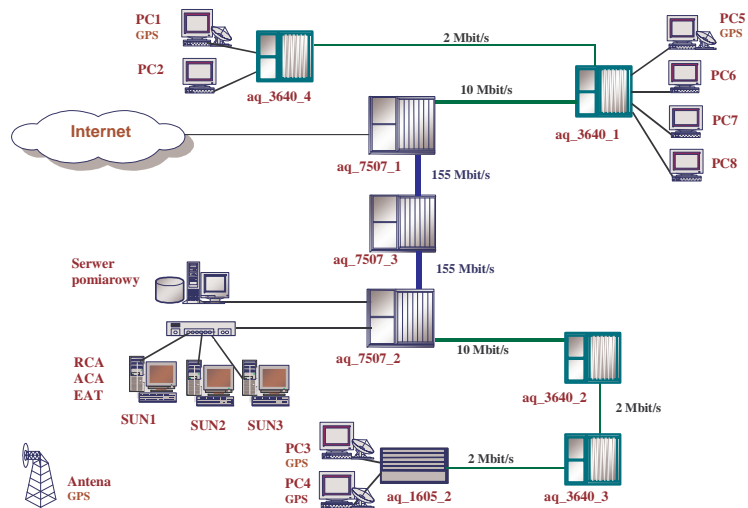
Rys.5. Ogólna architektura sieci AQUILA (RCA, ACA i EAToolkit tworzą warstwę RCL). RCA – agent zarządzania zasobami, ACA – agent sterowania przyjmowaniem nowych wywołań, EAToolkit – interfejs do komunikacji pomiędzy aplikacją i ACA, H – terminal użytkownika, ED – urządzenie brzegowe, CR – ruter szkieletowy, BR – ruter brzegowy



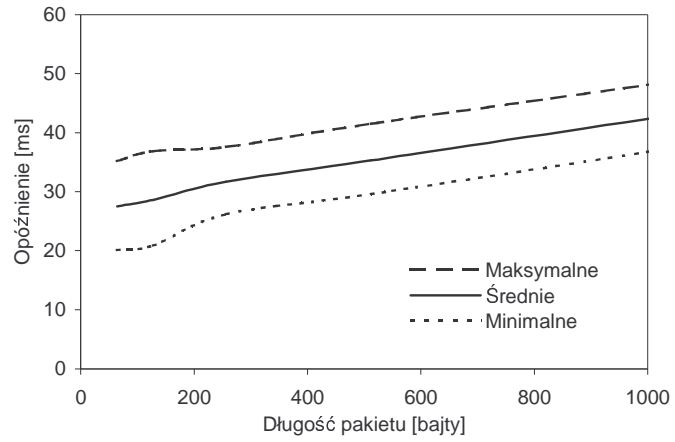
Rys.6. Architektura portu wyjściowego rutera. *TCL* – klasa ruchu, *WFQ* – algorytm *Weighted Fair Queueing*, *PQ* – algorytm *Priority Queueing*



Rys.7. a) Przykładowa struktura sieci IP i b) odpowiadająca jej hierarchia pul zasobów. CR – ruter szkieletowy, ED – urządzenie brzegowe, RP – pula zasobów, RL – liść w drzewie pul zasobów



Rys.8. Topologia sieci pilotowej IP QoS AQUILA. PC – terminal użytkownika (lub generator/analizator ruchu)



Rys.9. Pomiar opóźnienia przekazu pakietów przez sieć.



Rys.10. Ekran aplikacji RealPlayer uruchomionej w warunkach przeciążenia a) w sieci tradycyjnej i b) w sieci z jakością obsługi, z parametrem SBR = 264 kbit/s.