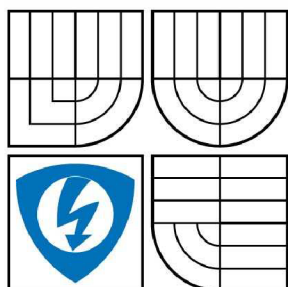


VYSOKÉ UČENÍ TECHNICKÉ V BRNĚ

BRNO UNIVERSITY OF TECHNOLOGY



**FAKULTA ELEKTROTECHNIKY A KOMUNIKACNÍCH
TECHNOLOGIÍ
ÚSTAV TELEKOMUNIKACÍ**

FACULTY OF ELECTRICAL ENGINEERING AND COMMUNICATION

DEPARTMENT OF TELECOMMUNICATIONS

**ANALÝZA SOUČASNÝCH TECHNOLOGIÍ ZAJIŠTĚNÍ
KVALITY SLUŽEB V PAKETOVÝCH SÍTÍCH**

BAKALÁŘSKÁ PRÁCE

BACHELOR'S THESIS

AUTOR PRÁCE

AUTHOR

VEDOUcí PRÁCE

SUPERVISOR

ANDREJ KULHÁNY

ING. JIŘÍ HOŠEK

BRNO 2008

LICENČNÍ SMLOUVA POSKYTOVANÁ K VÝKONU PRÁVA UŽÍT ŠKOLNÍ DÍLO

uzavřená mezi smluvními stranami:

1. Pan/paní

Jméno a příjmení:

Bytem:

Narozen/a (datum a místo):

(dále jen „autor“)

a

2. Vysoké učení technické v Brně

Fakulta elektrotechniky a komunikačních technologií

se sídlem Údolní 244/53, 602 00, Brno

jejímž jménem jedná na základě písemného pověření děkanem fakulty:

.....

(dále jen „nabyvatel“)

Čl. 1 Specifikace školního díla

1. Předmětem této smlouvy je vysokoškolská kvalifikační práce (VŠKP):

- disertační práce
- diplomová práce
- bakalářská práce
- jiná práce, jejíž druh je specifikován jako

.....

(dále jen VŠKP nebo dílo)

Název VŠKP:

Vedoucí/ školitel VŠKP:

Ústav:

Datum obhajoby VŠKP:

VŠKP odevzdal autor nabyvateli v*:

- tištěné formě – počet exemplářů
- elektronické formě – počet exemplářů

* hodící se zaškrtněte

2. Autor prohlašuje, že vytvořil samostatnou vlastní tvůrčí činností dílo shora popsané a specifikované. Autor dále prohlašuje, že při zpracovávání díla se sám nedostal do rozporu s autorským zákonem a předpisy souvisejícími a že je dílo dílem původním.
3. Dílo je chráněno jako dílo dle autorského zákona v platném znění.
4. Autor potvrzuje, že listinná a elektronická verze díla je identická.

Článek 2

Udělení licenčního oprávnění

1. Autor touto smlouvou poskytuje nabyvateli oprávnění (licenci) k výkonu práva uvedené dílo nevýdělečně užít, archivovat a zpřístupnit ke studijním, výukovým a výzkumným účelům včetně pořizování výpisů, opisů a rozmnoženin.
2. Licence je poskytována celosvětově, pro celou dobu trvání autorských a majetkových práv k dílu.
3. Autor souhlasí se zveřejněním díla v databázi přístupné v mezinárodní síti
 - ihned po uzavření této smlouvy
 - 1 rok po uzavření této smlouvy
 - 3 roky po uzavření této smlouvy
 - 5 let po uzavření této smlouvy
 - 10 let po uzavření této smlouvy(z důvodu utajení v něm obsažených informací)
4. Nevýdělečné zveřejňování díla nabyvatelem v souladu s ustanovením § 47b zákona č. 111/ 1998 Sb., v platném znění, nevyžaduje licenci a nabyvatel je k němu povinen a oprávněn ze zákona.

Článek 3

Závěrečná ustanovení

1. Smlouva je sepsána ve třech vyhotoveních s platností originálu, přičemž po jednom vyhotovení obdrží autor a nabyvatel, další vyhotovení je vloženo do VŠKP.
2. Vztahy mezi smluvními stranami vzniklé a neupravené touto smlouvou se řídí autorským zákonem, občanským zákoníkem, vysokoškolským zákonem, zákonem o archivnictví, v platném znění a popř. dalšími právními předpisy.
3. Licenční smlouva byla uzavřena na základě svobodné a pravé vůle smluvních stran, s plným porozuměním jejímu textu i důsledkům, nikoliv v tísní a za nápadně nevýhodných podmínek.
4. Licenční smlouva nabývá platnosti a účinnosti dnem jejího podpisu oběma smluvními stranami.

V Brně dne:

.....
Nabyvatel

.....
Autor

Abstrakt

Tato bakalářská práce je zaměřena na rozbor technologií pro zabezpečení kvality služeb v paketových sítích. V hlavní části se věnuji podrobnému rozboru technologií IntServ a DiffServ. Praktická část obsahuje návrh sítě v simulačním programu Openet Modeler. Navrhovaná síť je zaměřena hlavně na zabezpečení kvality služeb při přenosu hlasu pomocí technologie DiffServ. Při návrhu sítě jsem nasimuloval síť bez zabezpečení kvality služeb a síť se zabezpečením kvality služeb technologií DiffServ. Následně jsem porovnal výsledky simulací.

Klíčová slova

kvalita služeb, IP, IntServ, DiffServ, opnet modeler

Abstract

This bachelor thesis focus on analysis of technologies for guaranteed quality of services in packet network. In main part I devote to detailed study of technologies IntServ and DiffServ. Practical part of my bachelor thesis includes design of network in simulating program Openet Modeler. Designed network is specialized to guarantee quality of services during voice transmission by technology DiffServ. When designing the network I simulated guarantee of services in network without quality of service and in network with quality of services technology DiffServ. Finally I compared issues of simulations.

Key words

quality of service, IP, IntServ, DiffServ, opnet modeler

Prohlášení

Prohlašuji, že svou bakalářskou práci na téma „Analýza současných technologií zajištění kvality služeb v paketových sítích“ jsem vypracoval samostatně pod vedením vedoucího semestrálního projektu a s použitím odborné literatury a dalších informačních zdrojů, které jsou všechny citovány v práci a uvedeny v seznamu literatury na konci práce.

Jako autor uvedené bakalářské práce dále prohlašuji, že v souvislosti s vytvořením této práce jsem neporušil autorská práva třetích osob, zejména jsem nezasáhl nedovoleným způsobem do cizích autorských práv osobnostních a jsem si plně vědom následků porušení ustanovení § 11 a následujících autorského zákona č. 121/2000 Sb., včetně možných trestněprávních důsledků vyplývajících z ustanovení § 152 trestního zákona č. 140/1961 Sb.

V Brně dne.....

.....

Andrej Kulhány

Poděkování

Rád by jsem touto cestou poděkoval panu Ing. Jiřímu Hoškovi, vedoucímu této práce, za odbornou i lidskou podporu, trpělivost a připomínky k práci.

Také by jsem rád poděkoval celé své rodině za velkou morální podporu nejen při vypracování této práce, ale též po dobu celého studia.

V Brně dne.....

.....

Andrej Kulhány

Obsah

Obsah	8
Zoznam obrázkov.....	10
Zoznam tabuliek.....	12
Zoznam skratiek.....	13
Úvod.....	15
1. História a súčasnosť	16
2. QoS	18
2.1. Kvalita hlasu	19
2.1.1. Delay (oneskorenie)	19
2.1.2. Jitter (časové chvenie).....	20
2.1.3. Packet loss (stratovosť paketov)	20
2.1.4. Bandwidth (šírka prenosového pásma)	20
2.2. Popis najbežnejších typov aplikácií zameraných na prenos hlasu	21
2.2.1. Web	21
2.2.2. Multimediálne aplikácie.....	21
2.2.3. Aplikácie pracujúce v reálnom čase	21
3. Sieťová vrstva - protokol IP	22
3.1. Základné princípy metód pre zaistenie QoS na sieťovej vrstve.....	22
3.1.1. Best-Effort.....	22
3.1.2. Rezervovanie pásma	22
3.1.3. Priority	22
4. QoS technológie.....	23
4.1. Technológia ToS a DSCP	23
4.1.1. ToS (Type of Service).....	23
4.1.1. DSCP (Differentiated Service CodePoint).....	24
4.2. Technológia ATM.....	25
4.3. IntServ (Integrated Services)	26
4.3.1. Riadenie prístupu (Admission Control).....	27
4.3.2. RSVP (Resource reSerVation Protocol)	27
4.3.3. Správy PATH a RESV.....	27
4.3.4. Klasifikátor (Packet Classifier)	27
4.3.5. Plánovač paketov (Packet Scheduler).....	28

4.4. DiffServ (Differentiated Services).....	28
4.4.1. Triedenie a prispôsobenie premávky (Traffic conditioning)	29
4.5. Spolupráca technológií IntServ a DiffServ	31
5. OPNET Modeler	32
5.1. Základné prvky OPNET Modeler	32
5.1.1 Editor projektu (Project Editor)	33
5.1.2 Editor projektu – pracovná plocha.....	34
5.2. Návrh v OM	35
5.2.1 Vytvorenie sieťového modelu.....	35
6. Návrh vlastnej siete v OPNET Modeler	38
6.1. Vytvorenie topológie bez QoS.....	38
6.2. Nastavenie aplikácií	40
6.2.1 Application Config – Definície aplikácií.....	40
6.2.2 Profile Config – Profil aplikácie	41
6.3. Nastavenie aplikácií na jednotlivých uzloch.....	44
6.3.1 Nastavenie aplikácií na serveroch.....	44
6.3.1 Nastavenie spojenia klient – server.....	44
6.3.2 Nastavenie spojenia VoIP telefónov	46
6.3.3 Nastavenie Wi-fi siete.....	46
6.4. Nastavenie testovaných parametrov – štatistík.....	48
6.5. Nastavenie prevádzky na pozadie	49
6.6. Vytvorenie topológie s QoS.....	50
6.6.1 Profil WFQ (Weighted fair queueing)	54
6.6.2 Metódy RED/WRED	54
6.6.3 Nastavenie kvality služieb na smerovači	55
6.6. Nastavenie simulácie	57
6.7. Zobrazenie výsledkov simulácie.....	58
6.7. Výsledky simulácie	60
Záver	61
Zoznam literatúry a použitých zdrojov	62
Zoznam príloh	64

Zoznam obrázkov

Obr.1.1 Vývoj QoS.....	17
Obr.4.1 Štruktúra poľa ToS	23
Obr.4.2 Štruktúra poľa DiffServ	24
Obr.4.3 Referenčný model technológie IntServ	26
Obr.4.4 Doména DiffServ	29
Obr.4.5 Triedenie a prispôsobenie	29
Obr.4.6 Kombinácia služieb DiffServ a IntServ.....	31
Obr.5.1 Základná štruktúra editorov v OM.	33
Obr.5.2 Sieťový model v editore projektu.	33
Obr.5.3 Hlavné menu v editore projektu.	34
Obr.5.4 Tlačidlá projektového editoru.....	34
Obr.5.5 Textová oblasť editoru projektu.	35
Obr.5.6 Zobrazenie tipov v editore projektu.....	35
Obr.5.7 Menu project editor.....	36
Obr.5.8 Uzly	37
Obr.5.9 Link.....	37
Obr.6.1 Topológia siete bez_QoS	39
Obr.6.2 Objekt Application Config.	40
Obr.6.3 Nastavenie aplikácie FTP s vysokým zaťažením	41
Obr.6.4 Objekt Profile Config.	41
Obr.6.5 Nastavenie profilu provoz_voip.	43
Obr.6.6 Nastavenie podporovaných služieb na serveru FTP_server.	44
Obr.6.7 Nastavenie aplikácie Application: Destination Preference klienta Wkst_1.	45
Obr.6.8 Nastavenie aplikácie Application: Supported klienta Wkst_1.	45
Obr.6.9 Nastavenie atribútov Attributes telefónu VoIP_1_a.....	46
Obr.6.10 Nastavenie prístupového bodu Access Point Functionality wi-fi_router.	47
Obr.6.11 Príklad nastavenia globálnych (vľavo) a lokálnych (vpravo) parametrov.....	48
Obr.6.12 Nastavenie vlastnej prevádzky na pozadie.	49
Obr.6.13 Nastavenie preddefinovanej prevádzky na pozadie.....	50
Obr.6.14 Vytvorenie nového scenára s_QoS.....	50
Obr.6.15 Objekt QoS Config.	51
Obr.6.15 Nastavenie atribútov QoS Config.....	53

Obr.6.16 Metóda RED (vľavo) a metóda WRED (vpravo).....	55
Obr.6.17 Nastavenie atribútov Interface Information na smerovači.....	56
Obr.6.18 Nastavenie atribútov Traffic Classes na smerovači.....	56
Obr.6.19 Nastavenie atribútov Traffic Policies na smerovači.	57
Obr.6.20 Nastavenie atribútov oboch scenárov pre simuláciu.	58
Obr.6.21 Zobrazenie globálnej štatistiky voice, zmena omeškania (jitter).....	59
Obr.6.22 Zobrazenie globálnej štatistiky voice, oneskorenie koncových uzlov (delay) .	59

Zoznam tabuliek

Tab.1 Citlivosť rôznych typov služieb na parametre siete.....	20
Tab.2 Služby na jednotlivých klientoch.....	47
Tab.3 zaradenie front a priradená váha.....	53

Zoznam skratiek

AAL – Adaptation Layer

AP – Access Point

ATM – Asynchronous Transfer Mode

BA – Behavior Aggregate

DiFFServ – Defferentiated Services

DSCP – Differentiated Service CodePoint

FIFO – First In – First Out

FTP – File Transfer Protocol

HTML – Hyper Text Markup Language

HTTP – Hypertext Transfer Protocol

Ipv4 – Internet Protocol version 4

Ipv6 – Internet Protocol version 6

ID – Identity Protocol

IETF – Internet Engineering Task Force

IntServ – Integrated Services

IP – Internet Protocol

ISDN – Integrated Services Data Network

ISP – Internet Service Provider

MAC address – media access control adress

MF – Multifield

OPNET – Optimum Network Performance

OM – Opnet Modeler

PSTN – Public Switched Telephone Network

QoS – Quality of Service

RED – Random Early Detection

RESV – Reservation Request message

RSVP – Resource Reservation Protocol

SVC – Switched Virtual Channel

TCP – Transmission Control Protocol

ToS – Type of Service

UDP – User Datagram Protocol

VoIP – Voice over Internet Protocol

WFQ – Weighted fair queueing

WRED – Weighted Random Early Detection

WWW – World Wide Web

XML – Extensible Markup Language

Úvod

V posledných rokoch sa dôležitosť Quality of Service (QoS) či kvality služieb, pre paketové siete rapídne zvýšila. Dnes je kvalita služieb jednou z ústredných súčastí celkového paketového prenosu. Existujú rôzne typy riešení a návrhov architektúr, ktoré zavádzajú úrovne kvality služieb a zoradujú dátové jednotky prichádzajúcej premávky podľa priorít. Kvalita služieb môže byť implementovaná v referenčnom modeli v rôznych vrstvách. Najčastejšie sa používa implementácia na úrovni linkovej vrstvy pri použití technológie ATM, alebo na úrovni IP, ktorá pracuje na sieťovej vrstve. Táto bakalárska práca sa zameriava na používané technológie pre zaistenie kvality služieb QoS v paketových sieťach. Práca sa venuje stručnej histórii kvality služieb, rozdelení používaných technológií pre zaistenie kvality služieb a koniec práce je venovaný návrhu ako aj realizácii modelu siete v programe OPNET.

1. História a súčasnosť

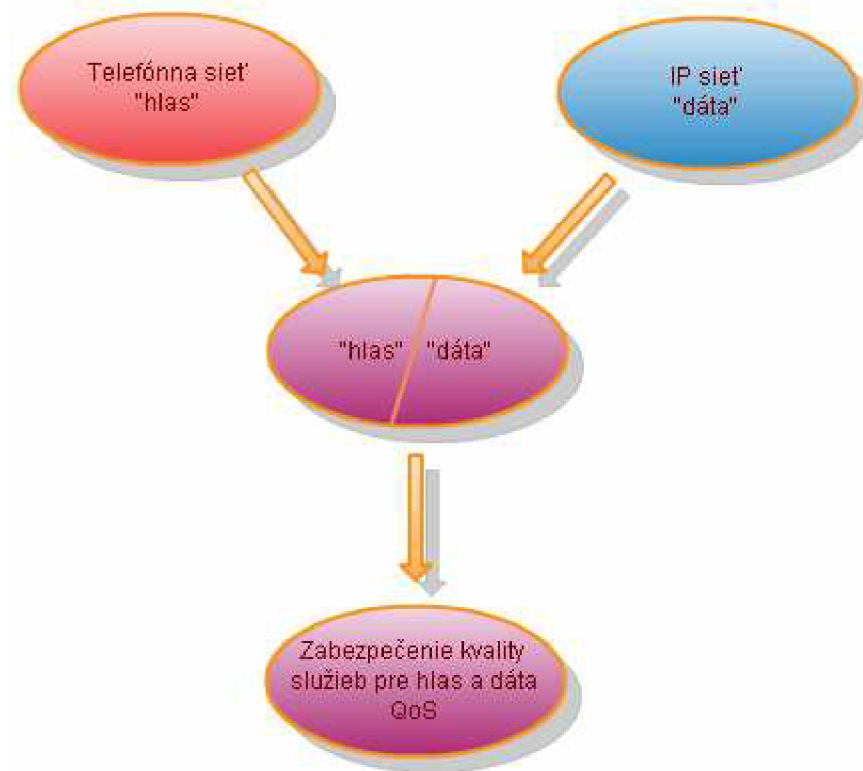
Ako názov tejto kapitoly napovedá, táto časť zhodnocuje nedávnu históriu telekomunikačných sietí. Na počiatkoch telekomunikačných sietí boli dve oddelené siete, jedna pre prenos hlasu a druhá pre prenos dát. Každá sieť začínala jednoduchým cieľom prenosu špecifického typu informácií.

V prvých telefónnych sieťach boli dve hlavné kritériá kvality. Prvé, pravdepodobnosť straty, čo je pravdepodobnosť, že pokus o spojenie môže byť zablokovaný kôli nedostupnosti spojovacieho okruhu. Druhé, hlasová kvalita, ktorá závisela na prenosovej kvalite behom volania ako aj strata prenosu, šum v obvode, echo, atď. Originálna telefónna sieť bola preto navrhnutá tak, aby bolo zaručených dost okruhov, ktoré poskytnú pravdepodobnosť strát napríklad 1%. Telefónna sieť bola ďalej navrhnutá na tzv. sieť end to end („spojenie koniec s koncom“) s prenosovým plánom optimalizovaným pre hlas tak, že zhoršenie kvality spojenia, ako straty, šum, echo a oneskorenie, boli prijateľné.

Hlas sa samozrejme prenáša v reálnom čase. Skorá IP sieť bola úplne iným typom siete ako telefónna sieť. Predovšetkým bola navrhnutá k tomu aby prenášala dáta, ktoré by mohli byť uložené v sieti a doručené neskôr. Aj keď hlas a dáta majú zreteľne rôzne charakteristiky dopravy, rokom 1990 sa dve oddelené siete začali zlučovať do jednej. A tak sa začala snaha o konvergovanie („zjednotenie“) týchto dvoch typoch sietí. V týchto časoch bola predstava o zlúčení pre hlas a dáta ako abstraktný pojem, no dnes už nikto nemôže pochybovať o zjednotení prenosu hlasu a dát. Vývoj dospel až ku technológii QoS (Obr.1.1), ktorá sa snaží zaistiť potrebné parametre daného spojenia [1].

Podstatná časť dnešných existujúcich telefónnych sietí je založená na princípe PSTN (Public Switched Telephone Network) sietí. Čo znamená, že hovor si rezervuje spojenie medzi dvoma užívateľmi a nik iný toto spojenie nemôže využívať. Až keď je hovor ukončený, linka je voľná pre ostatných užívateľov. V Internetovom telefonovaní, známom ako VoIP (Voice over IP) alebo IP telefónia je prenos uskutočnený v rámci IP paketovej siete. Je tu možné bez rezervácie spojenia posilať pakety medzi dvoma a viacerými stranami. Dosiahneme to digitalizáciou audio signálov, zaznamenaných použitím mikrofónu, IP telefónu alebo počítača, implementovaných do paketov a potom poslaním po sieti použitím internetových protokolov.

Doručovateľ potom tieto pakety poskladá a prehrá pôvodnú správu pomocou reproduktorov. Iné typy médií ako napríklad video, alebo zdieľané aplikácie je tiež možné zahrnúť do prenosu bez výraznejších zásahov a zmien. V tejto metóde je kladená veľká náročnosť na zabezpečenie kvality služby QoS, keďže nie je žiaden spôsob ako garantovať, že pakety dorazia k cieľu.



Obr.1.1 Vývoj QoS [1].

2. QoS

V dnešnej dobe, keď sa v telekomunikačnom prostredí nasadzujú riešenia konvergovaných sietí zlučujúcich hlas, dáta a video do jedného logického paketového celku, je nutné sa zamyslieť nad požiadavkami týchto zložiek. Prenos hlasu cez IP sieť skrátene VoIP (Voice over Internet Protocol) získava stále viac na dôležitosti. O úspechu technológie VoIP sa dnes nedá pochybovať, predsa však existujú isté požiadavky a predpoklady, ktoré musia byť splnené nato aby bola zaručená požadovaná kvalita služby QoS (Quality of Service). QoS je modul, ktorý riadi tok dát v sieti a tým zaisťuje správne delenie rýchlostí a nedochádza tak k zahlcovaniu siete. Dnešný užívateľ požaduje, aby mu bola poskytnutá určitá kvalita, ktorú tento modul dostatočne zaisťuje pozdĺž celej cesty v telekomunikačnej sieti.

Vo všeobecnosti, QoS poskytuje lepšie a viac predvídateľné sieťové služby tak, že:

- zlepšujú charakteristiky strát
- formujú sieťovú prevádzku
- nastavujú prioritu sieťovej prevádzky naprieč celou sieťou
- zabraňujú a manažujú sieťové zahltenie

Nutné body pre zavedenie QoS:

- oprávnenosť žiadosti o sieťové zdroje
- rozlíšenie a označenie typu služieb v termináloch a predovšetkým v prepojujúcich uzloch
- rozdelenie dátových tokov do prioritných front
- ochrana pred preťažením front
- obsluha front podľa priority
- riešenie vzniku chýb pri prenose
- rozloženie záťaže vzhľadom na optimalizáciu výkonu siete
- zaistenie spoľahlivosti a bezpečnosti služieb
- spoplatnenie služieb

Zo zavedenia QoS do existujúcich alebo nových sietí plynú hlavne tieto výhody:

- vyššia výkonnosť siete
- väčšia flexibilita a kontrolovateľnosť
- rýchlejšie nasadzovanie aplikácií a služieb
- redukcia nákladov na prevádzku

2.1. Kvalita hlasu

Pojem kvalita hlasu zahŕňa subjektívne hodnotenie akosti rečového signálu, preneseného komunikačným kanálom. Pojem ďalej popisuje degradáciu spôsobenú prenosom a zohľadňuje aspekty ľudského vnímania rečového signálu. Meranie kvality prenosu hlasu sa stáva jednou z mála platforiem, pomocou ktorej je možné navzájom porovnať odlišné prenosové technológie, ktoré sú vo svojej podstate najbližšie z pohľadu jednotlivých koncových užívateľov, to všetko vďaka zvyšujúcej sa zložitosti a komplexnosti sietí. Konkrétne elementy QoS architektúry závisia od typu prenášaných informácií (hlas, dáta, alebo video).

Pre VoIP definuje QoS obmedzenia špecifické k prenosu hlasu ako sú delay (oneskorenie), jitter (časové chvenie), packet loss (stratovosť paketov) a bandwidth (šírka prenosového pásma). Výsledná kvalita hlasového prenosu závisí na množstve parametrov, ktoré zahŕňajú okamžité fyzikálne vlastnosti všetkých prvkov, ktoré tvoria komunikačný reťazec.[2].

2.1.1. Delay (oneskorenie)

Oneskorenie vyjadruje dobu potrebnú k prenosu paketu od odosielateľa k príjemcovi. Je najdôležitejšia pri interaktívnych aplikáciách. V prípade prenosu väčšieho množstva údajov či vizuálnych informácií, oneskoreniu až tak veľmi neprekáža pokiaľ je tok dát plynulý. V prípade interaktívnych aplikácií (video, audio, on-line hry, atď.) však je potrebné mať odozvu čo najmenšiu. Doba odozvy sa udáva v milisekundách (ms).

2.1.2. Jitter (časové chvenie)

Časové chvenie, alebo zmena omeškania udáva oneskorenie jednotlivých paketov počas prenosu údajov. Matematicky je vyjadrená ako derivácia oneskorenia, jednoduchšie povedané hovorí ako sa s časom oneskorenie mení. Táto vlastnosť je dôležitá z pohľadu na poradie prichádzajúcich paketov keďže v praxi sa stáva, že pakety odoslané neskôr prídu do cieľa skôr ako pakety, ktoré boli odoslané skôr. Časové chvenie má negatívny dopad najmä na multimedialne prenosi. Udáva sa v mikrosekundách (μ s).

2.1.3. Packet loss (stratovosť paketov)

Strata paketov značí hodnotu koľko percent paketov nedorazí od odosielateľa k príjemcovi, teda koľko paketov sa po ceste stratí. Udáva sa v percentách (%).

2.1.4. Bandwidth (šírka prenosového pásma)

Definuje objem dát v bitoch prenesených za jednotku času. Je kritická predovšetkým u multimedialných aplikáciách a prenose väčšieho množstva dát, ale veľký význam má aj pri obyčajnom „surfovaní“. V súčasnosti je už väčšina prenosových liniek vybavená dostatočnou kapacitou, problémy tu vznikajú v úzkom hrdle, kde sa prenosové pásmo značne degraduje (niektoré firewally, proxy servery, smerovače a podobne). Udáva sa v bitoch, kilobitoch a megabitoch za sekundu (b/s, kb/s, Mb/s).

Tab.1 Citlivosť rôznych typov služieb na parametre siete [9].

<i>Služba</i>	<i>Citlivosť na oneskorenie</i>	<i>Citlivosť na časové chvenie</i>	<i>Citlivosť na stratovosť paketov</i>	<i>Citlivosť na šírku pásma</i>
<i>Hlas</i>	vysoká	vysoká	stredná	veľmi nízka
<i>e-mail</i>	nízka	nízka	vysoká	nízka
<i>Prehliadanie webu</i>	vysoká	nízka	vysoká	stredná
<i>Prenos súboru</i>	nízka	nízka	stredná	vysoká
<i>Videokonferencia</i>	vysoká	vysoká	stredná	vysoká

2.2. Popis najbežnejších typov aplikácií zameraných na prenos hlasu

2.2.1. Web

Trend v komunikáciách, ktorý je používaný hlavne na prehliadanie stránok, ale aj na prenos súborov, bussines aplikácie, školenia, prehrávanie hudby či videa a pod. Protokol http, ktorý je pre prístup k webovým serverom používaný, využíva transportný protokol TCP (Transmission Control Protocol) a zaisťuje teda spoľahlivý prenos. Aplikácie ako sú elektronická pošta, webové prehliadače, prenos súborov fungujú dobre aj bez technológie QoS, kedy majú všetky aplikácie rovnakú prioritu pri odosielaní svojich dátových paketov. Na druhej strane pre multimedialne aplikácie ako je už spomínané VoIP (telefonovanie cez Internet) či interaktívne hry je zaručenie kvality prenosu bez QoS nedosiahnuteľné pre spokojnosť zákazníka.

2.2.2. Multimedialne aplikácie

Z predchádzajúceho textu vyplýva, že multimedialne aplikácie sa neobídu bez nejakej formy modulu QoS, sú totiž veľmi citlivé na oneskorenie a na šírku pásma. Streaming hlasu a videa je technológia šírenia audiovizuálneho obsahu po Internete. Streaming predstavuje spôsob kontinuálneho doručovania dát smerom od poskytovateľa hlasu, alebo video záznamu smerom k zákazníkovi, je to takzvaný synchronný prenos dát. Takto je možné zaistiť živé vysielanie, užívateľ totiž môže sledovať streamované dáta už behom prenosu. Kvalita streamu je automaticky prispôbovaná rýchlosti internetového pripojenia. Systém streaming hlasu a videa nie je príliš náchylný na časové chvenie (jitter) [3].

2.2.3. Aplikácie pracujúce v reálnom čase

Príkladom aplikácie pracujúcej v reálnom čase je prenos hlasu a videa pomocou konferencie. Tieto aplikácie sú citlivé na kvalitu oneskorenia a rýchlosť doručenia.

3. Sieťová vrstva - protokol IP

Dáta sa v IP (Internet Protocol) posielajú po blokoch nazývaných datagramy, ktoré putujú sieťou nezávisle. Na začiatku komunikácie nie je potreba nadväzovať spojenie či inak „pripravovať“ cestu dátam. Protokol IP je najrozšírenejším komunikačným protokolom v súčasných počítačových sieťach, ktorý sa používa tak ako v podnikových sieťach, tak aj v sieťach ISP (Internet Service Provider). Na ňom je založená aj najväčšia celosvetová sieť Internet. Snahu o dosiahnutie čo najlepšej kvality služby môžeme rozdeliť do nasledujúcich metód.[4].

3.1. Základné princípy metód pre zaistenie QoS na sieťovej vrstve

3.1.1. Best-Effort – Internet bol v začiatkoch jeho vývoja navrhnutý pre Best-Effort systém ako sme sa mohli dozvedieť v prvej časti nazvanej „História a súčasnosť“. Ide o systém bez garancie doručenia paketov. Aj v dnešnej dobe Internetu je tento systém doručovania najrozšírenejší.[2].

3.1.2. Rezervovanie pásma – základ tejto metódy je v rezervácií požadovaného pásma, ktoré je potrebné na prenos aplikácie. Po ukončení prenosu sa rezervované pásmo uvoľní. Typickým príkladom je technológia IntServ (Integrated Services) s protokolom RSVP (Resource Reservation Protocol) [2].

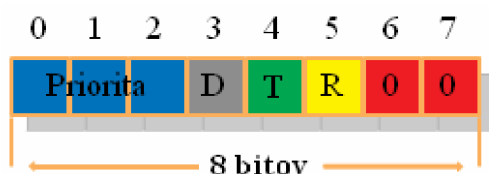
3.1.3. Priority – sú aplikované niekoľkými metódami na rôznych úrovniach komunikácie. Ide o spôsob definície na základe akých kritérií bude priorita prevedená. Typickým príkladom sú technológie DiffServ (Differentiated Services) [2].

4. QoS technológie

4.1. Technológia ToS a DSCP

4.1.1.ToS (Type of Service)

ToS predstavuje spôsob ako môžeme rozlíšiť pakety do jednotlivých tried. Type of Service je osembitové pole nazvané u IPv4 (Internet Protocol version 4). Type of Service (IP ToS), ktorého zobrazenie môžeme vidieť na Obr.4.1. Toto osembitové pole sa skladá z troch bitov, ktoré určujú prioritu podľa špecifikácie typu služby, troch bitov určujúcich požiadavky na prenos a dvoch nepoužívaných bitoch, ktoré sú rezervované pre možnosť potreby v budúcnosti (preto majú nastavené hodnoty nula). Pôvodne bolo pole ToS určené na spracovanie paketov v smerovači, avšak nebolo až tak využívané. Neskôr začalo byť toto pole využívané pre podporu technológie DffServ. U IPv6 je pridelená značka vložená do triedy provozu TC (Traffic Class) [5], [6].



Obr.4.1 Štruktúra poľa ToS [5], [6].

Vyznačená modrá plocha zahŕňa prvé tri bity, ktoré určujú typ priority pri spracovaní paketu smerovačom. Bolo definovaných osem úrovní IP priorít (precedence), kde vyššia hodnota znamená vyššiu prioritu [8].

- Network Control (7)	111	- Flash (3)	011
- Internetwork Control (6)	110	- Immediate (2)	010
- Critical (5)	101	- Priority (1)	001
- Flash Override (4)	100	- Routine (0)	000

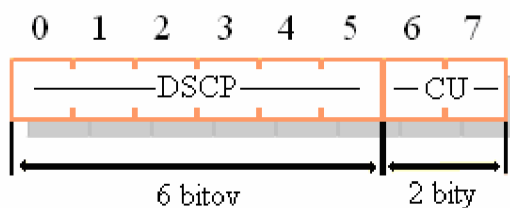
Ďalšie tri bity vyznačené šedou (D), zelenou (T) a žltou (R) vyjadrujú nasledovne:

- 3. bit (D) Delay (oneskorenie)
 - Normálne oneskorenie 0
 - Malé oneskorenie 1
- 4. bit (T) Throughput (priepustnosť)
 - Normálna priepustnosť 0
 - Vysoká priepustnosť 1
- 5. bit (R) Reliability (spoľahlivosť prenosu)
 - Normálna spoľahlivosť prenosu 0
 - Vysoká spoľahlivosť prenosu 1

Z kombinácií týchto šiestich bitov a pridaním dvoch núl pre šiesty a siedmy bit vzniká značka, ktorá môže vyzeráť napr. nasledovne: 11101000 ma prioritu (7) Network Control, normálne oneskorenie, vysokú priepustnosť a normálnu spoľahlivosť prenosu.

4.1.1. DSCP (Differentiated Service CodePoint)

Ak je smerovač používaný pre DiffServ ako uzol diferencovaných služieb, tých istých osem bitov poľa ToS (Type of Service) alebo TC (Traffic Class) sú nahradené poľom diferencovaných služieb. Obr.4.2 znázorňuje náhradu poľa ToS a TC poľom diferencovaných služieb. Z ôsmich bitov poľa DiffServ, prvých šesť bitov je používaných pre označenie paketov DiffServ, pričom tieto bity sú označované ako kódové body DSCP a posledné dva bity predstavujú bity pre ďalšie použitie [5], [6].



Obr.4.2 Štruktúra poľa DiffServ [5], [6].

4.2. Technológia ATM

ATM (Asynchronous Transfer Mode) je technológia s nadväzovaným spojovaním, preto musí byť pred každým dátovým prenosom medzi dvoma koncovými uzlami nadviazané a ustanovené virtuálne spojenie. Konkrétne bola ATM technológiou určená pre širokopásmové ISDN (Integrated Services Data Network) a pre integráciu telekomunikačných služieb do jednotnej siete, avšak kvôli zložitosti a tým aj vysokej cene tejto technológie sa nepresadila v očakávanej miere, aj napriek tomu perspektívne vlastnosti do budúcnosti ju stále udržujú pri živote v aplikáciách spojených s interaktívnymi službami.

Najväčšou výhodou ATM oproti ďalším komerčne používaným technológiám je rozsiahla podpora zaistení kvality služieb QoS pre jednotlivé spojenia. QoS na ATM je dané typom služby s rezervovaním pásma prostredníctvom PVC (Permanent Virtual Channel), nastavovaný manuálne, alebo SVC (Switched Virtual Channel), nastavovaný dynamicky a rušený pomocou signalizačných protokolov. O prispôbenie užívateľsky nezávislú ATM vrstvu vyššej aplikačnej vrstvy sa stará adaptačná vrstva AAL (Adaptation Layer).

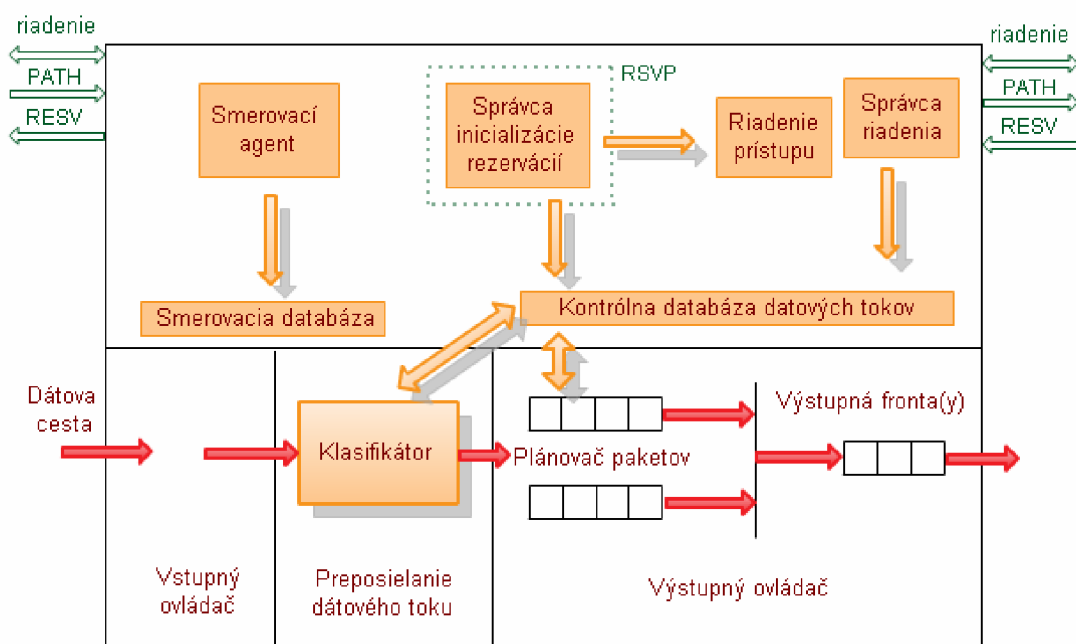
AAL definuje služby pre vyššie vrstvy, z čoho vyplýva, že táto vrstva je priamo závislá na využívaných službách a tak poskytuje pre rôzne typy služieb rôzne druhy funkcií. Adaptačná vrstva rieši prispôbenie ATM vrstvy k užívateľským službám, ktoré predstavujú veľmi širokú škálu rôznych požiadavkou a AAL vrstva musí byť schopná splniť všetky tieto požiadavky. Pre zaistenie čo najrýchlejším spôsobom splniť všetky potrebné požiadavky bolo definovaných päť rôznych typov adaptačných vrstiev. Tie sú označené skratkou AAL a číslovaním od 1 až 5 (AAL1, AAL2,... ,AAL5).

Pre zameranie tejto bakalárskej práce je zaujímavý typ AAL2, ktorý sa používa pre komprimované prenosi zvuku a obrazu. Koncové zariadenie môže požadovať špecifickú kvalitu prenosu QoS. Jednotlivé typy prenosov sú totiž odlišné a majú protichodné požiadavky. Prenos hlasu a videa v reálnom čase požaduje definované oneskorenie s minimálnym rozptylom. Naproti tomu dátové prenosi nemajú presne dané požiadavky na minimálne pásmo prenosu, oneskorenie, na druhú stranu vyžaduje veľkú vyrovnávaciu pamäť pre efektívny prenos [2].

4.3. IntServ (Integrated Services)

Pri integrovaných službách (IntServ), aplikácia oznámi počítačovej sieti svoje požiadavky na prenos dát. Z toho plynie, že IntServ definuje určité vlastnosti, ktoré by pri prenose paketov mali byť dodržané. Následne počítačová sieť overí, či môže týmto požiadavkám vyhovieť. Pokiaľ nie je možné vyhovieť, aplikácia môže znížiť svoje požiadavky na QoS, alebo odložiť prenos. Ak počítačová sieť požiadavkám vyhovie, musí informovať všetky komponenty (napr. smerovače v uzloch počítačovej siete), aby mohli pre dané spojenie rezervovať zodpovedajúci objem prostriedkov. Zpravidla sa rezervuje určitá veľkosť vyrovnávacej pamäti pre tvorbu front vo vnútri smerovača, určitá šírka pásma pre spojenie medzi dvoma smerovačmi, apod. K tomuto účelu slúži rezervačný protokol RSVP (Resource reSerVation Protocol).

Týmto spôsobom, rezervácie dostatočných prostriedkov môžeme pre každý dátový tok v sieti zaistiť (aspoň teoreticky) požadovanú kvalitu služby. Hlavnou nevýhodou tohto prístupu je, že väčšina zariadení, cez ktoré prechádzajú IP pakety dátových tokov a pre ktoré je potreba zistiť určitú kvalitu služby, musí podporovať protokol RSVP. Vo väčších sieťach by sme narazili na vysoké pamäťové a výkonnostné nároky na smerovače. Model IntServ našiel uplatnenie hlavne pri zaistovaní QoS pre aplikácie ako videokonferencie, prenos hlasu v IP sieťach, ktoré vyžadujú včasné doručenie dát [6], [9]. Na Obr.4.3 je znázornený referenčný model technológie IntServ.



Obr.4.3 Referenčný model technológie IntServ [6], [9].

Pre správnu funkciu integrovaných služieb musia byť v smerovačoch a hostiteľoch (príjemca a odosielateľ QoS) implementované nasledujúce komponenty:

4.3.1. Riadenie prístupu (Admission Control)

Na žiadosť aplikácie sú podľa dostupnosti sieťových prostriedkov akými sú, požadovaná vyrovnávacia pamäť a šírka pásma tieto prostriedky buď poskytnuté a spätne zaslané pozitívne potvrdenie, alebo zamietnuté a vrátená negatívna odpoveď. Dá sa teda povedať, že riadenie prístupu realizuje rozhodovací algoritmus, či k novému toku môže byť udelená rezervácia.

4.3.2. RSVP (Resource reSerVation Protocol)

Signalizačný protokol, ktorý je určený pre rezerváciu sieťových prostriedkov pomocou správ PATH a RESV. Tento protokol je nutný k vytvoreniu a udržovaniu stavov v koncových zariadeniach a v smerovačoch pozdĺž celého toku dát.

4.3.3. Správy PATH a RESV

V architektúre QoS založenej na protokole RSVP sú dva základné prvky, zdroje a ciele. Na oboch prvkoch je nutné aby bežali RSVP procesy, ktoré sa podieľajú na vymieňaní RSVP správ svojich hostiteľov. V podstate si vymieňajú dva druhy správ, PATH a RESV správy. Zdroj služby pošle správu PATH, keď je táto správa prijatá príjemcom služby a tento príjemca chce urobiť rezerváciu pre daný RSVP tok, odpovedá RESV správou, ktorá je poslaná spätne k odosielateľovi po rovnakej ceste ako správa PATH.

Koncová rezervácia je úspešne zarezervovaná, keď odosielateľ prijme správu RESV. Ak zdroj služby pošle správu PATH a príjemca tejto správy neposkytne rezerváciu, vygeneruje sa RESV ERROR správa, ktorá je taktiež spätne odoslaná rovnakou cestou k odosielateľovi. RSVP využíva smerovacie tabuľky v smerovačoch k určeniu cesty k určitému cieľu.

4.3.4. Klasifikátor (Packet Classifier)

Klasifikátor, či triedič paketov je súčasťou smerovačov aj koncových zariadení, ktorý sa stará o efektívne riadenie dátovej dopravy a to tak, že každý prichádzajúci paket je namapovaný klasifikátorom do určitej triedy. So všetkými paketmi, ktoré boli zaradené do rovnakej triedy, bude v plánovači paketov zachádzané rovnako. Voľba týchto tried je

založená na zdrojovej a cieľovej IP adrese a čísla portu (alebo na ďalších hodnotách), ktoré musia byť pridané ku každému paketu [6], [9].

4.3.5. Plánovač paketov (Packet Scheduler)

Riadi odosielanie paketov používaním súborových front. Plánovač paketov musí byť implementovaný v mieste, kde sú pakety zaraďované do front.

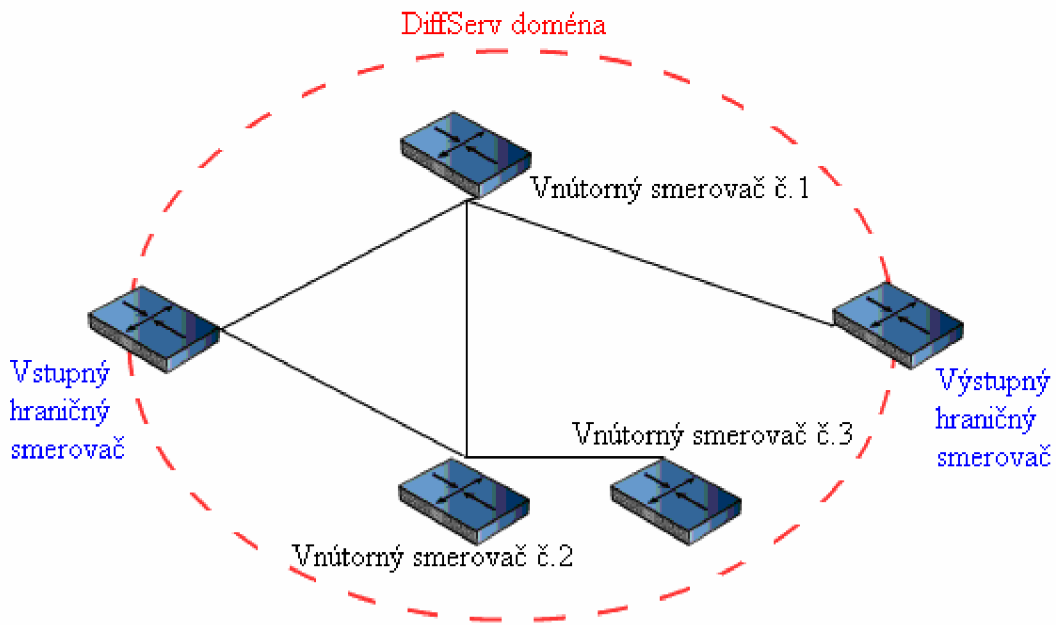
4.4. DiffServ (Differentiated Services)

Klasické uzly siete väčšinou využívajú princíp FIFO „First In – First Out“. To značí, že dátová jednotka, ktorá dorazila do uzlu najskôr, je taktiež skôr spracovaná. Spracované dátové jednotky sú potom predávané do výstupných front a postupne odosielené. Pre riadenie kvality služieb bola vyvinutá rada riešení, z ktorých sa najviac rozšírila technológia diferencovaných služieb (DiffServ). Táto architektúra je založená na jednoduchom modeli, kde sú dátové toky združené do tried podľa rovnakého typu služby. IP pakety vstupujúce do počítačovej siete, sú označené značkou, ktorá hovorí ako sa má s paketmi zaobchádzať, respektívne určuje jeho triedu prenosu. Toto označenie paketov prebieha iba na vstupe do počítačovej siete. Pomocou tejto značky smerovače riadia cestu každého paketu.

DiffServ poskytne zaistenie kvality služieb na základe klasifikácie paketov do rôznych tried s podobnými vlastnosťami. K roztriedeniu paketov do jednotlivých tried sa využíva technológia ToS (viď.Kap.4.1.1). Triediť pakety môžeme podľa rôznych hodnôt:

- Adresa IP
- MAC adresa
- Identifikátor portu prijímača
- Typ aplikácie (identifikovaný podľa čísel portov TCP/UDP)

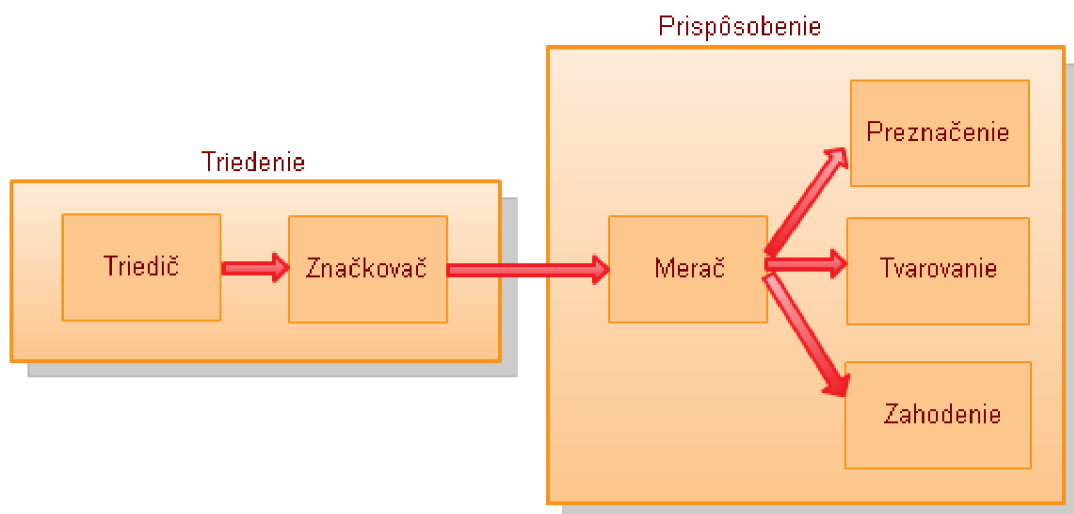
Oblasti, v ktorých je s triedami zachádzané rovnakým spôsobom sa nazývajú DiffServ domény. Uzly siete, ktoré zabezpečujú komunikáciu s prostredím mimo domény sa nazývajú hraničné (viď. Obr.4.4). Ich úlohou je zaradiť každú dátovú jednotku do príslušnej triedy. Klasifikácia paketov, či označenie značkami prebieha na vstupnom smerovači a to ako pri vstupe paketov do siete, tak aj pri prechode z jednej DiffServ domény do inej.



Obr.4.4 Doména DiffServ [7].

4.4.1. Triedenie a prispôsobenie premávky (Traffic conditioning)

U technológii DiffServ sú za rozdelenie paketov do tried zodpovedné okrajové smerovače. Funkcie, ktoré plnia okrajové smerovače, sú často označované ako triedenie a prispôsobenie riadenia premávky. Triediaci modul obsahuje triedič a značkovač. Triedič vyberie paket podľa vopred definovaných pravidiel. Výstup z triediacej jednotky je nasledovne vedený do bloku prispôsobenia premávky (viď. Obr.4.5).



Obr.4.5 Triedenie a prispôsobenie [8].

Triedenie – Triedič delí prichádzajúci tok paketov na základe obsahu položky DSCP (Vid'. Kap.4.1.2) v hlavičke paketu do niekoľkých skupín podľa preddefinovaných pravidiel. Sú definované dva základné typy triedičov:

- Behavior Aggregate (BA)
- Multifield (MF)

Medzi najjednoduchšie DiffServ triediče patrí BA triedič, ktorý vyberá pakety iba podľa hodnoty DSCP v hlavičke paketu. Triedič BA sa používa vtedy, keď už je DSCP určené (paket už je označený).

MF triedič používa kombinácie jedného alebo viacej polí v hlavičke IP paketu (zdrojová adresa, cieľová adresa, zdrojový port, cieľový port, protokol ID "Identity Protocol"). Z toho vyplýva, že zložitejší triedič MF môže byť veľmi flexibilný, oproti tomu BA triedič je oveľa jednoduchší a môže byť realizovaný ako priame vyhľadávanie založené na hodnote DSCP. Akonáhle je dokončené triedenie, triedič posielajú pakety do bloku prispôsobenia na ďalšie spracovanie [8], [9].

Prispôsobenie – Dátový tok vybraný triedením je ďalej vedený k bloku prispôsobenia, kde dôjde k meraniu dátového toku. Meranie je využívané k meraniu parametrov prichádzajúceho dátového toku od klientov a tieto hodnoty sú prevedené so zmluvnými parametrami. Tie pakety, ktoré vyhovujú zmluvným parametrom sú vpustené do vnútra siete a tie pakety, ktoré prekračujú zmluvné parametre sú ďalej upravované podľa triedy, do ktorej boli zaradené. Postupy, ktoré blok prispôsobenia môže prijať zahŕňa preznačovanie (remarker), tvarovanie premávky (traffic shaper) a zahadzovanie paketov (dropper).

Preznačovanie (Remarker) – Proces pri ktorom dôjde preznačeniu paketov, ktoré už boli označené. K tomuto procesu môže dôjsť napríklad na rozhraní dvoch administratívnych domén, ktoré využívajú rôzne DSCP.

Tvarovanie premávky (Traffic shaping) – Inteligentná metóda, ktorá sa stará o nasledovné: pakety, ktoré vybočujú zo zadefinovaných podmienok sú smerovačom pozastavené, pričom sa mení priebeh okamžitého objemu odosielaných dát v čase v porovnaní s prijímaných dát. Priebeh sa vyrovnáva tým, že dáta prichádzajúce na smerovač nepravidelnou rýchlosťou, sú odosielané stálou priemernou rýchlosťou prijímaných dát.

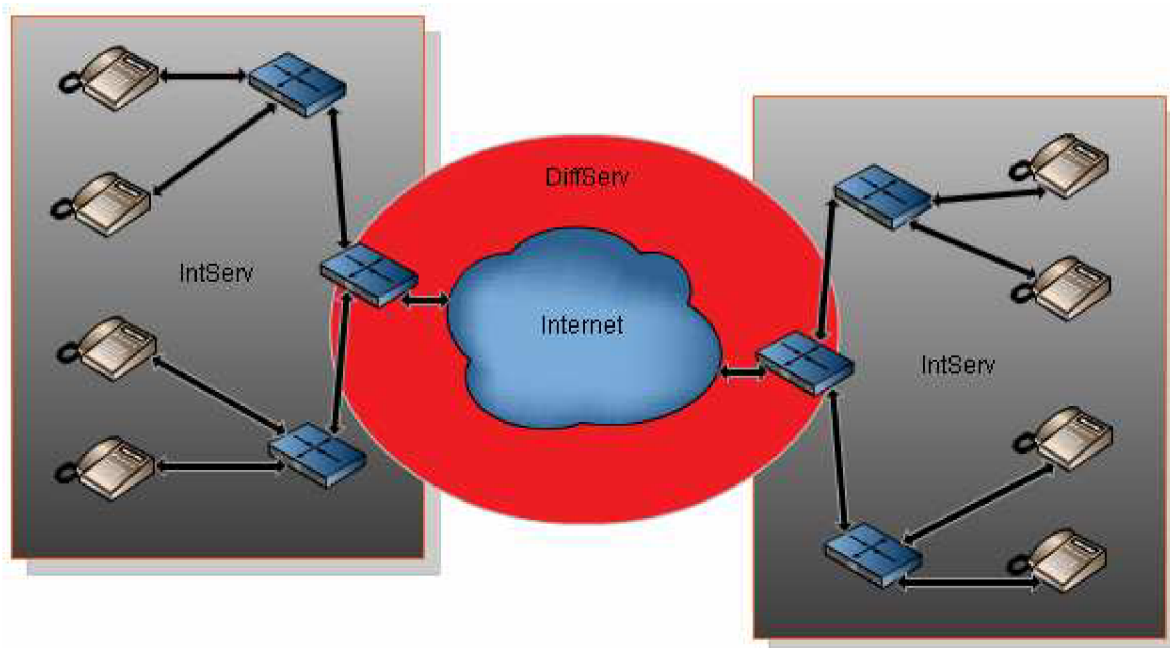
Zahadzovanie (Dropper) – Pri prekročení dohodnutých podmienok dôjde k zahodeniu paketov. K tomuto procesu môže dôjsť napríklad vyčerpaním kapacity front, alebo prekročením určitého objemu prichádzajúcich dát.

4.5. Spolupráca technológií IntServ a DiffServ

Organizácia IETF (Internet Engineering Task Force, česky „Komise techniky Internetu“), ktorá sa zaoberá štandardizáciou Internetu navrhla pre zaistenie QoS v IP sieťach dva hlavné modely. Prvým navrhnutým bol model Integrovaných služieb (IntServ) a druhý model Diferencovaných služieb (DiffServ).

Stručne môžeme povedať, že technológia IntServ pristupuje k riadeniu kvality služieb iným spôsobom ako technológia DiffServ. IntServ pracuje na princípe rezervácie pásma, DiffServ rozdeľuje všetky prenášané dátové jednotky do tried, ktoré priradujú rôznym dátovým jednotkám prislúchajúce priority.

Technológia DiffServ je využívaná vo vnútri infraštruktúr sietí (tzv. chrbticová infraštruktúra sietí), naopak technológia IntServ sa využíva v prístupových sieťach (tzv. koncové siete). Použitím technológie IntServ pre nasadenie v sieťach s veľkým zaťažením nie je vhodné, pretože silne zaťažuje sieťové prvky. Najlepšie výsledky sa dosahujú kombináciou služieb DiffServ a IntServ, ktorá je zobrazená na nasledujúcom obrázku (viď.Obr.4.6) [14].



Obr.4.6 Kombinácia služieb DiffServ a IntServ [14].

5. OPNET Modeler

Opnet Modeler (OM) patrí do celkového softwarového balíku OPNET (Optimum Network Performance) od americkej firmy OPNET Technologies. Tento software predstavuje simulačné vývojové prostredie, vhodné pre návrh, simuláciu a analýzu sietí. Veľmi dôležitou vlastnosťou OM je jeho široká možnosť tvorby rôznych štatistík z danej simulácie. Program umožňuje simulovať akúkoľvek architektúru sietí. Efektívnosť a rýchlosť práce v ňom umožňuje jeho grafické prostredie v ktorom je široká možnosť tvorby rôznych štatistík z danej simulácie. Výsledok určitej štatistiky môžeme generovať do správy vo formáte XML (Extensible Markup Language) alebo HTML (Hyper Text Markup Language), alebo dáta uložiť do tabuliek.

Simulácia prebieha s určitým zrýchlením, takže je možné spracovať týždenný chod siete v niekoľkých hodinách nasimulovaného času [15].

5.1. Základné prvky OPNET Modeler

Opnet Modeler má niekoľko základných prvkov:

- **subnet – podsieť** (spojené stanice, servery, routre apod.)
- **node model – model uzlu** (model stanice, serveru, routra apod.)
- **process model – model procesu** kde sú definované jednotlivé procesy modelu uzlu (napr. Vysielanie riadiacich informácií apod.)

Editory:

OM využíva objektovo orientovaného programovania a grafických editorov s aktuálnymi štruktúrami sietí. OM je “jednoduchý“ hierarchický editor, ktorý presne popisuje spojenie štruktúr reálnej siete a protokolov. Je tvorený z troch základných editorov (Obr.5.1) [15].

- **Project Editor** – editor projektu,
- **Node Editor** – editor uzlu,
- **Process Editor** – editor procesu.

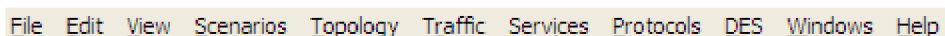
Z dôvodu zamerania tejto bakalárskej práce bude potrebné podrobnejšie rozobranie editoru projektu. Keďže s editorom uzlu (predstavuje nižšiu úroveň ako editor projektu) a editorom procesu (rozhranie najnižšej úrovne) sa pri návrhu nestretáme.

5.1.2 Editor projektu – pracovná plocha

V okne editoru projektu, (viď Obr.5.2) je niekoľko oblastí, ktoré sú dôležité pre výstavbu prevedenia modelu [15]:

- hlavné menu,
- tlačidlá,
- textová oblasť,
- tipy.

Hlavné menu – Každý editor má hlavné menu, ktoré editoru projektu ukazuje názorné rozloženie jednotlivých ponúk (viď. Obr.5.3).



Obr.5.3 Hlavné menu v editore projektu.


Tlačidlá – Pod hlavným menu sa nachádzajú tlačidlá, respektíve buttons. Pomocou nich môžeme rýchlo aktivovať vybranú funkciu editoru. Obr.5.4 ukazuje rozloženie tlačidiel pre editor projektu.

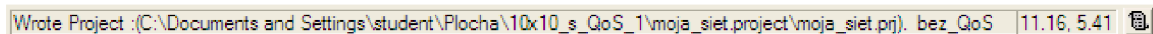


Obr.5.4 Tlačidlá projektového editoru.

Vysvetlenie tlačidiel:

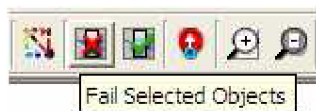
- | | |
|-------------------------------|---|
| 1. nový, | 11. vloženie topológie z VNE serveru, |
| 2. otvoriť, | 12. otvoriť okno pre sledovanie dátových tokov, |
| 3. uložiť, | 13. generovanie správy a rozdielov scenárov, |
| 4. tlačiť, | 14. konfigurovať jednotlivé udalosti simulácie a spustenie simulácie, |
| 5. otvoriť paletu objektov, | 15. ukázať výsledok simulácie, |
| 6. chybné vybraný objekt, | 16. skryť alebo zobrazit' panely grafov, |
| 7. otvoriť zhody odkazu, | 17. generovať sumár. |
| 8. späť k rodičovkej podsieti | |
| 9. zväčšiť náhľad, | |
| 10. zmenšiť náhľad, | |

Textová oblasť – Je umiestnená dole v okne modeleru. Sprostredkováva informácie o stave prevedenej akcie. Ďalej sa táto ikona  využíva pre zistenie poslednej prevedenej operácie (viď. Obr.5.5).



Obr.5.5 Textová oblasť editoru projektu.

Tipy – Pokiaľ ukážeme kurzorom na tlačidlo alebo na položku menu, zobrazí sa nám bublina s kontextovou nápovedou (viď. Obr.5.6).



Obr.5.6 Zobrazenie tipov v editore projektu.

5.2. Návrh v OM

5.2.1 Vytvorenie siet'ového modelu

Pri vytváraní nového projektu je potrebné najprv vytvoriť nový projekt spoločne so scenárom (pozadím). Dôležité je, že projekt môže obsahovať viac na sebe nezávislých scenárov a je možné medzi nimi prepínať, čo bude definované pri vytváraní vlastnej siete [15].

V prvotnom vytváraní projektu bude definovaná:

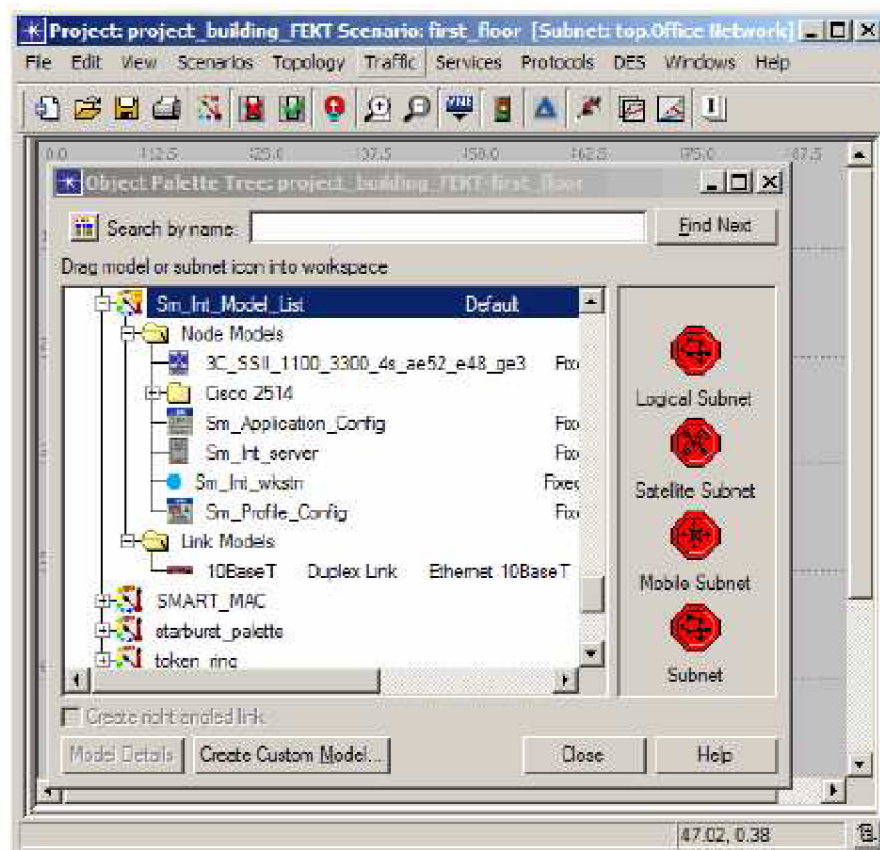
- Topológia a veľkosť siete,
- Mapa, ktorá sa zobrazí na pozadie scenára,
- Objekty, s ktorými budeme pracovať (Object Palette).

Ďalším krokom bude vytvorenie už spomínaného vlastného projektu:



1. Spustiť OPNET Modeler 12.0 .

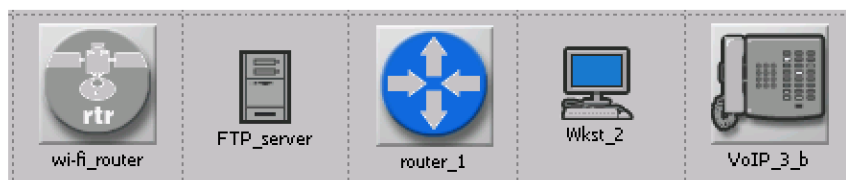
2. Vybrať položku **File > New...** a označenie výberu nového projektu (**project**).
3. Zadať meno projektu a meno scenára (meno projektu: **siet_Ic**, a meno scenára: **bez_QoS**).
4. Zadať **Create empty scenario** pre vytvorenie prázdneho scenára.
5. Vybrať položku **Office** a nechať zatrhnutú hodnotu **Use metric unit**.
6. Zadať požadované rozmery plochy **100 x 100 meters**.
7. Vybrať objekty, ktoré budú používané **Sn_Int_Model_List** a dokončiť s tlačением tlačidla **Finish** (viď Obr.5.7).



Obr.5.7 Menu project editor[15].

Sieťový model sa vytvára v projektovom editore, kde máme k dispozícii dva typy objektov:

Node – (uzol) zastupuje reálne objekty ako napr. smerovače, prepínače, rozbočovače, servery, pracovné stanice apod. (viď Obr.5.8).




Obr.5.8 Uzly .

Link – zastupuje prenosové média pre spojenie jednotlivých uzlov. K dispozícii sú ako elektrické prenosové média, tak aj optické prenosové média. Ikona tohto objektu je uvedená na Obr.5.9.



Obr.5.9 Link .

K vybraniu jednotlivých objektov slúži ikona  **Object Palette**.

8. Prejsť k vytvoreniu topológie siete. OM umožňuje aj rýchlu konfiguráciu návrhu siete tzv. **Rapid Configuration**, ktorú je možné nájsť v ponuke **Topology – Rapid Configuration** (čo nie je tento prípad).

6. Návrh vlastnej siete v OPNET Modeler


Postup pri návrhu siete:

- Vytvoriť topológiu bez zaistenia kvality služieb (QoS), sieť *bez_QoS*.
- Nastaviť testované parametre.
- Vytvoriť topológiu so zaistením kvality služieb (QoS), sieť *s_QoS*.
- Nastaviť simuláciu.
- Zobrazíť výsledky simulácií.

6.1. Vytvorenie topológie bez QoS

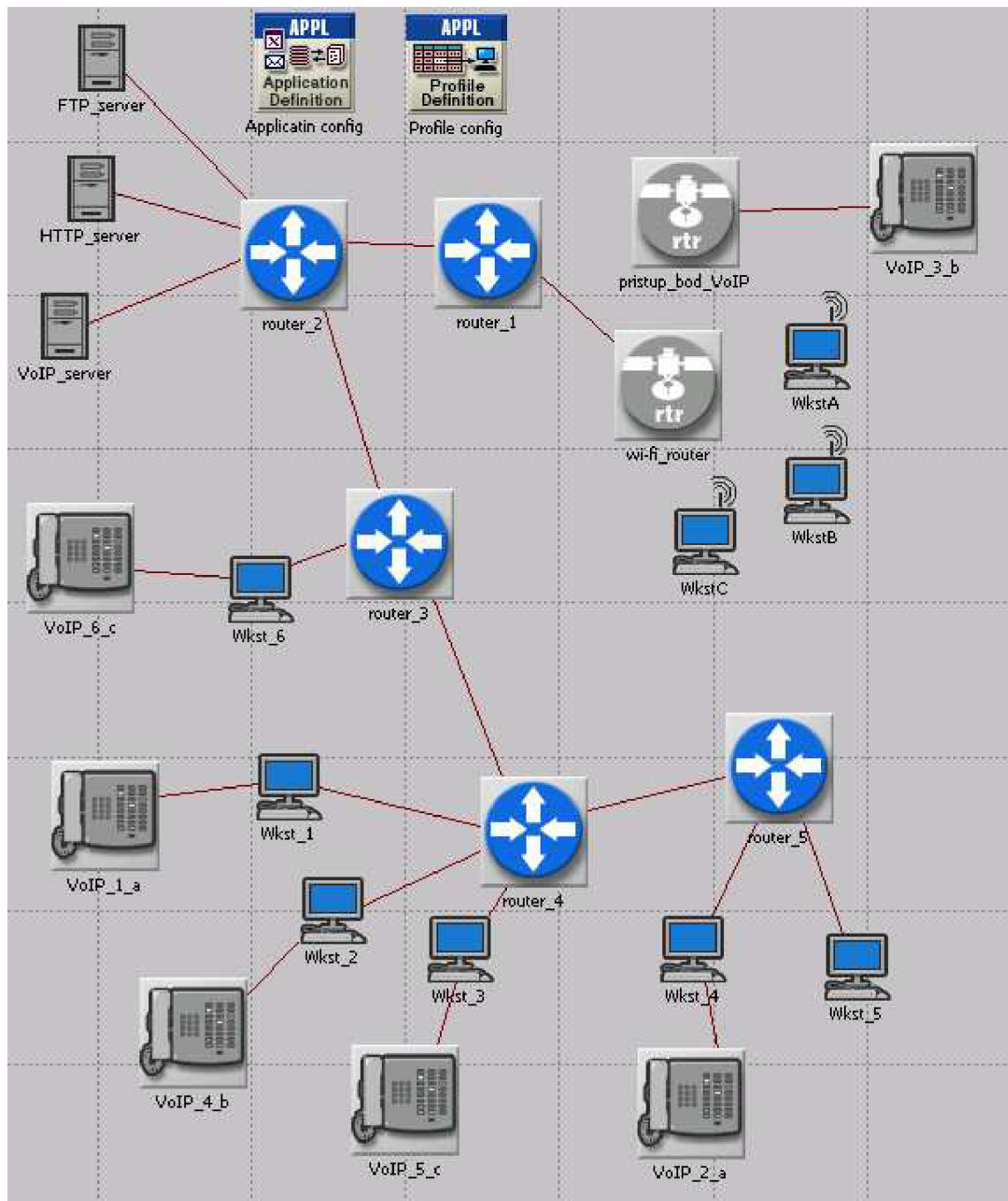
Najskôr si v OM vytvoríme nový projekt (viď.Kap.5.2.1) a na jeho plochu umiestnime tieto objekty:

- *Application Config* (viď.Kap.6.2.1),
- *Profile Config* (viď.Kap.6.2.2).

Objekty vložíme na plochu po kliknutí na ikonu  *Object Palette – Configure Palette* a vyberieme objekty, ktoré potrebujeme na vytvorenie siete nazvanej *siet_1c* so scenárom *bez_QoS*:

- *ethernet_server_adv* – model serveru (server),
- *BN_ASN_4s_e4_f2_fe4_sl12_tr4* – model smerovača (router),
- *wlan_ethernet_router* – model bezdrôtového smerovača (wireless router),
- *Compaq_Personal_Workstation_500* – model klienta (workstation),
- *wlan_wkstn_adv* – model bezdrôtového klienta (wireless workstation),
- *ip-phone* – model VoIP telefónu (VoIP phone).

Nasledovne boli jednotlivé objekty postupne poprepájané sieťovou technológiou 100BaseT a 10BaseT. Pomocou objektu z položky *Object Palette – 100BaseT* boli poprepájané jednotlivé smerovače, servery so smerovačmi a klienti so smerovačmi. Pomocou *Object Palette – 10BaseT* boli poprepájané VoIP telefóny s príslušnými klientmi. Vytvorená topológia siete **bez_QoS** (Obr.6.1).



Obr.6.1 Topológia siete bez_QoS .

6.2. Nastavenie aplikácií

Každá aplikácia je v OM tvorená dvoma objektmi a to už spomínanými *Application Config* a *Profile Config*.

6.2.1 Application Config – Definície aplikácií

Aplikácie, ktoré budú v sieti je nutné definovať, na to slúži objekt *Application Config*, ktorý už máme vložený v našom návrhu siete (viď. Obr.6.2).



Obr.6.2 Objekt Application Config.

Aby sme mohli editovať jednotlivé aplikácie, musíme kliknúť pravým tlačidlom na objekt *Application Config* a z kontextového menu zvoliť *Edit Attributes*. Ďalej budeme editovať položku s názvom *Application Definitions*. Nachádza sa tu položka *rows*, ktorá udáva maximálny počet nami definovaných aplikácií. V položke *name* definujeme meno aplikácie. Meno by malo čo najlepšie vystihovať danú aplikáciu. Môžeme mať napr. viac *FTP* aplikácií, preto vytvorenú aplikáciu treba patrične odlišiť označením napr. *FTP High Load*. Aby sme mohli definovať podrobnejšie chovanie aplikácií, klikneme na položku *description*.

Pre štandardné aplikácie ako *FTP*, *HTTP* apod. sú už preddefinované hodnoty ako napr. *Low Load*, *Medium Load* a *High Load*, čoho využijeme pri nastavení. Zo širokej možnosti aplikácií (*Database*, *Email*, *FTP*, *HTTP*, *Print*, *Remote Login*, *Video Conferencing*, *Voice*), ktoré majú určité preddefinované hodnoty budeme používať aplikáciu *FTP*, *HTTP*, *Voice*.

Všetky aplikácie majú primárne vypnuté hodnoty, takže keď chceme nastaviť aplikáciu prenosu súborov pomocou protokolu FTP (File Transfer Protocol), musíme kliknúť pravým tlačidlom na objekt *Application Config* a z kontextového menu zvoliť *Edit Attributes*. Ďalej budeme editovať položku s názvom *Application Definitions*, kde si nájdeme *File Transfer (Heavy)* a v položke *Description* nastavíme *High Load*, čo predstavuje vysoké zaťaženie aplikácie (viď. Obr.6.3).

Attribute	Value
name	Applicatin config
Application Definitions	(...)
rows	16
row 0	Database Access (Heavy),(...)
row 1	Database Access (Light),(...)
row 2	Email (Heavy),(...)
row 3	Email (Light),(...)
row 4	
Name	File Transfer (Heavy)
Description	(...)
Custom	Off
Database	Off
Email	Off
Ftp	High Load
Http	Off
Print	Low Load Medium Load
Remote Login	High Load
Video Conferencing	Edit...
Voice	Off
row 5	File Transfer (Light),(...)
row 6	File Print (Heavy),(...)

Obr.6.3 Nastavenie aplikácie FTP s vysokým zaťažením .

Nasledovne nastavíme aj aplikáciu **Web Browning (Heavy HTTP 1.1)** čo predstavuje službu WWW(World Wide Web) pomocou HTTP protokolu „surfovanie po Internete“. V položke **Description** nastavíme rovnako vysoké zaťaženie aplikácie **High Load**. Rovnakým postupom zopakujeme aj nastavenie pre poslednú použitú aplikáciu v našom návrhu, ktorou je **Voice over IP Call (PCM Quality)** a teda prenos hlasu cez IP protokol.

6.2.2 Profile Config – Profil aplikácie

Definovali sme si aplikácie a čaká nás nastavenie profilov, aplikáciám sa tu priradia hodnoty, podľa ktorých napr. budú aplikácie spúšťané. Pre definovanie profilu aplikácie sa používa objekt **Profile Config**, ktorý už máme vložený v návrhu siete (Obr.6.4) [9].



Obr.6.4 Objekt Profile Config.

Rovnakým spôsobom ako pri nastavení *Application Config* zvolíme položku *Edit Attributes*. Zvolíme položku *rows*, ktorá nám udáva počet profilov. Počet profilov sa nemusí rovnať počtu aplikácií. Po zadaní *rows* hodnotou **3** sa nám zobrazia tri profily, každý nami vytvorený profil má nasledujúce položky [9]:

- *Profile Name* – meno profilu,
- *Application* – vložíme aplikáciu, ktorú sme vytvorili v objekte Application config:
 - *Rows* – počet aplikácií, ktoré budú spustené s týmto profilom,
 - *Name* – meno aplikácie, ktoré sme vytvorili v objekte Application config,
 - *Start Time Offset (seconds)* – nastavenie času kedy sa daná aplikácia spustí. Ak nastavíme napr. hodnotu 0, aplikácia sa spustí ihneď po začiatku profilu, takže tento parameter závisí na voľbe Start Time profilu,
 - *Duration* – počet opakovaní a časový odstup k ďalšiemu zopakovaniu,
 - *Reapeatability* – udáva počet opakovaní a časový odstup k ďalšiemu zopakovaniu,
- *Operation Mode* – definuje ako sa budú dané aplikácie spúšťať:
 - *Serial (Ordered – prikázaný)* – môže byť spustená najskôr jedna a potom ďalšia aplikácia, poradie je presne dané pozíciou aplikácie v zozname,
 - *Serial (Random – náhodný)* – poradie je náhodné,
 - *Simultaneous* – všetky aplikácie štartujú v rovnaký čas,
- *Start Time (sec)* – definuje, kedy bude profil spustený,
- *Duration (sec)* – definuje dobu trvania profilu,
- *Reapeatability* – udáva počet opakovaní a časový odstup k ďalšiemu zopakovaniu

Pri položke *Start Time Offset* je možné nastaviť buď, aby aplikácie začínali v presne stanovený okamžik voľbou typu *constant*, alebo napr. štart v určitom časovom rozsahu, ktoré je vyjadrené pravdepodobnostnou funkciou.

Ďalej je tu veľké množstvo nastavení položky *Start Time Offset – Distribution name* napr. (poisson, geomteric, lognormal, bernoulli atď.). Rovnomerné rozloženie pravdepodobností, ktoré je využívané v našom simulačnom modeli, je nastavené funkciou *Distribution name = uniform*.

Máme zvolené tri profily, pre ukážku si popíšeme nastavenie profilu, ktorý sme si pomenovali *provoz_voip*, do záložky *Application* vložíme nami definovanú aplikáciu, ktorú sme si vytvorili, pre tento profil je to aplikácia *Voice over IP Call (PCM Quality)*. V položke *Start Time Offset – Distribution name* nastavíme rovnomerné rozloženie pravdepodobnosti *uniform*. nastavenie je zobrazené na nasledujúcom obrázku (Obr.6.4). Následne v záložke *Operation Mode* nastavíme *Simultaneous* čo značí, že všetky aplikácie budú spustené v rovnaký čas. Z toho vyplýva, že položka *Start Time (sec)* musí byť nastavená rovnako pre všetky profily (aj pre *provoz_ftp*, *provoz_http*). Doba trvania, teda *Duration (sec)* sme stanovili do konca simulácie, takže zvolíme *End of Simulation*. Celkové nastavenie profilu je zobrazené na nasledujúcom obrázku (Obr.6.5).

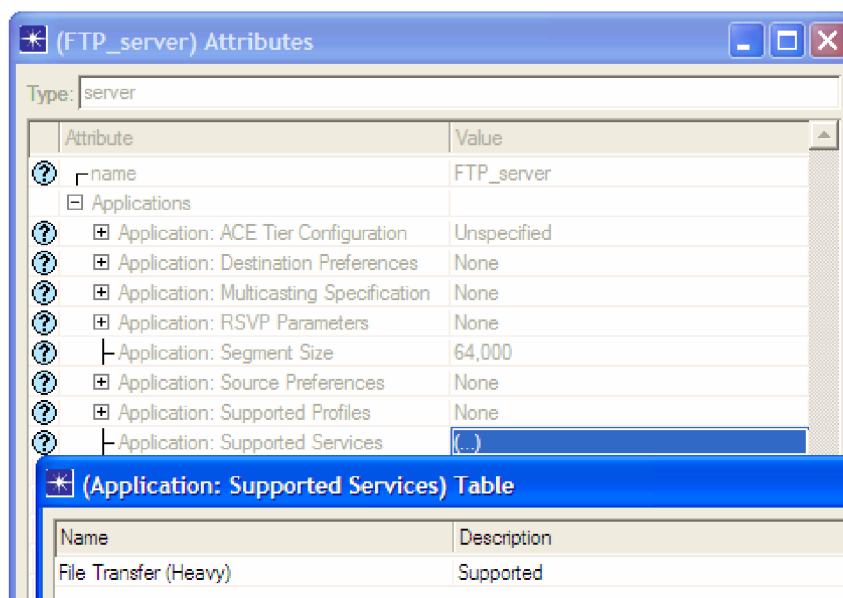
Attribute	Value
name	Profile config
Profile Configuration	(...)
rows	3
row 0	provoz_ftp,(...),Simultaneous,constant (1...
row 1	provoz_http,(...),Simultaneous,uniform (1...
row 2	
Profile Name	provoz_voip
Applications	(...)
rows	1
row 0	
Name	Voice over IP Call (PCM Quality)
Start Time Offset (seconds)	uniform (5, 15)
Duration (seconds)	End of Profile
Repeatability	(...)
Inter-repetition Time (sec...)	exponential (300)
Number of Repetitions	Unlimited
Repetition Pattern	Serial
Operation Mode	Simultaneous
Start Time (seconds)	uniform (10, 15)
Duration (seconds)	End of Simulation
Repeatability	Once at Start Time

Obr.6.5 Nastavenie profilu *provoz_voip*.

6.3. Nastavenie aplikácií na jednotlivých uzloch

6.3.1 Nastavenie aplikácií na serveroch

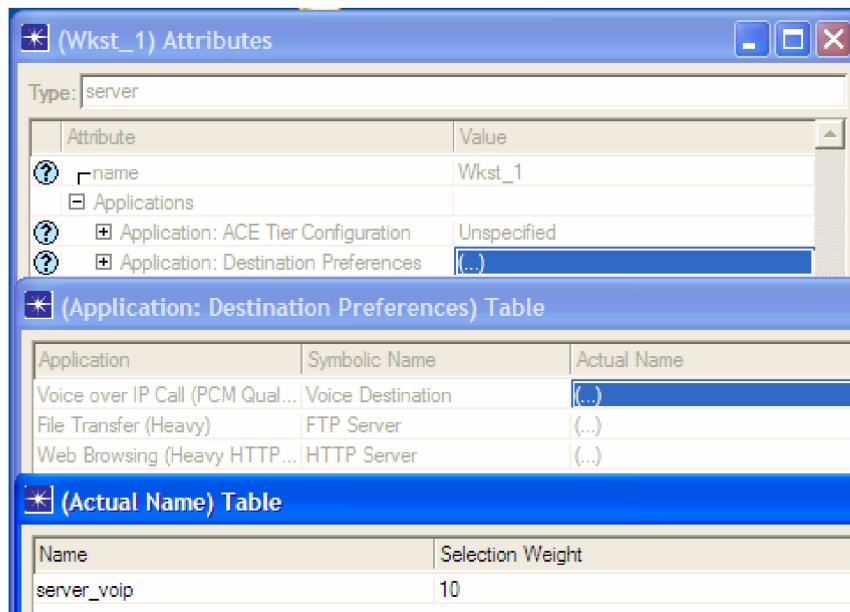
Keďže sme nastavili definície a profily aplikácií, môžeme prejsť k nastaveniu na jednotlivých *nodes* (uzloch). Začneme s nastavením serverov, ktoré nastavíme všetky rovnakým postupom, opäť si pre ukážku ukážeme jeden postup a to nastavenie serveru pre podporu aplikácií FTP. Objekt máme už vložený na pracovnej ploche. Ako prvé nastavíme meno serveru, takže do položky *Name* vložíme názov serveru *FTP_server*. Prejdeme na položku *Application: Supported Services* a nastavíme meno našej podporovanej služby FTP a kolonku *Description* nastavíme na *Supported* čiže podporované (Obr.6.6) [9].



Obr.6.6 Nastavenie podporovaných služieb na serveru FTP_server.

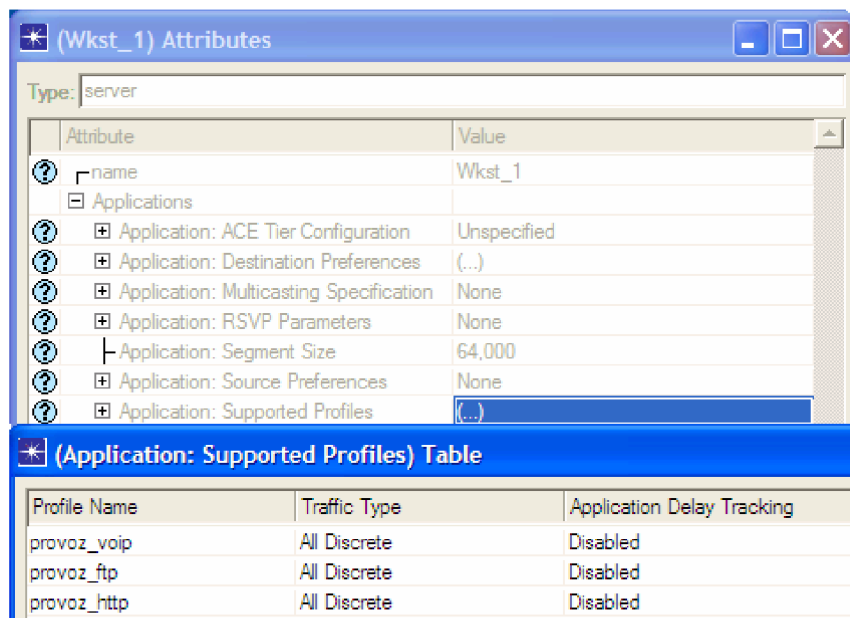
6.3.1 Nastavenie spojenia klient – server

Ak chceme nastaviť konkrétne spojenie medzi serverom a klientom, je nutné nastaviť hodnoty v kolonke *Application: Destination Preference*. Ak ich nenastavíme, OM zvolí náhodne vybraný server a klient v sieti a prevedie na nich zvolené procesy. Pre výber serveru slúži položka *Selection Weight*. Príklad nastavenia spojenia klient – server môžeme vidieť na obr. 6.7, ktorý zobrazuje klienta *Wkst_1*, na ktorom sú spustené všetky tri aplikácie, čo znamená, že je nastavené spojenie so všetkými servermi.



Obr.6.7 Nastavenie aplikácie Application: Destination Preference klienta Wkst_1.

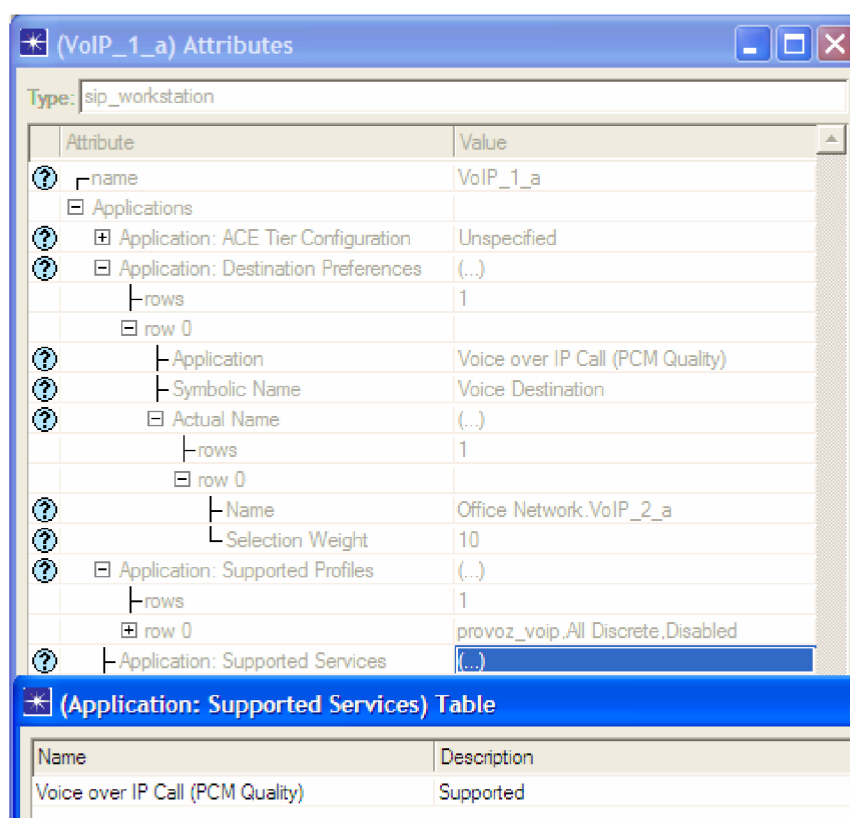
Nasledovne nastavíme v položke **Application: Supported Profiles** profily, ktoré budú na klientovi bežať. Z čoho vyplýva, že pre náš ilustrovaný príklad klienta **Wkst_1** je potrebné nastavenie všetkých troch profilov (viď.Obr.6.8).



Obr.6.8 Nastavenie aplikácie Application: Supported klienta Wkst_1.

6.3.2 Nastavenie spojenia VoIP telefónov

Volanie medzi VoIP účastníkmi sme nastavili nasledovne. V položke *Attribute* je nutné pre každý telefón zvoliť jednoznačné meno napr. *VoIP_1_a*, tento telefón bude volať s účastníkom označeným ako *VoIP_2_a*, *VoIP_3_b* s účastníkom *VoIP_4_b* a účastník *VoIP_5_c* s účastníkom *VoIP_6_c*. Pre ukážku zobrazujeme na nasledujúcom obrázku nastavenie Attributes telefónu *VoIP_1_a* (Obr.6.9).

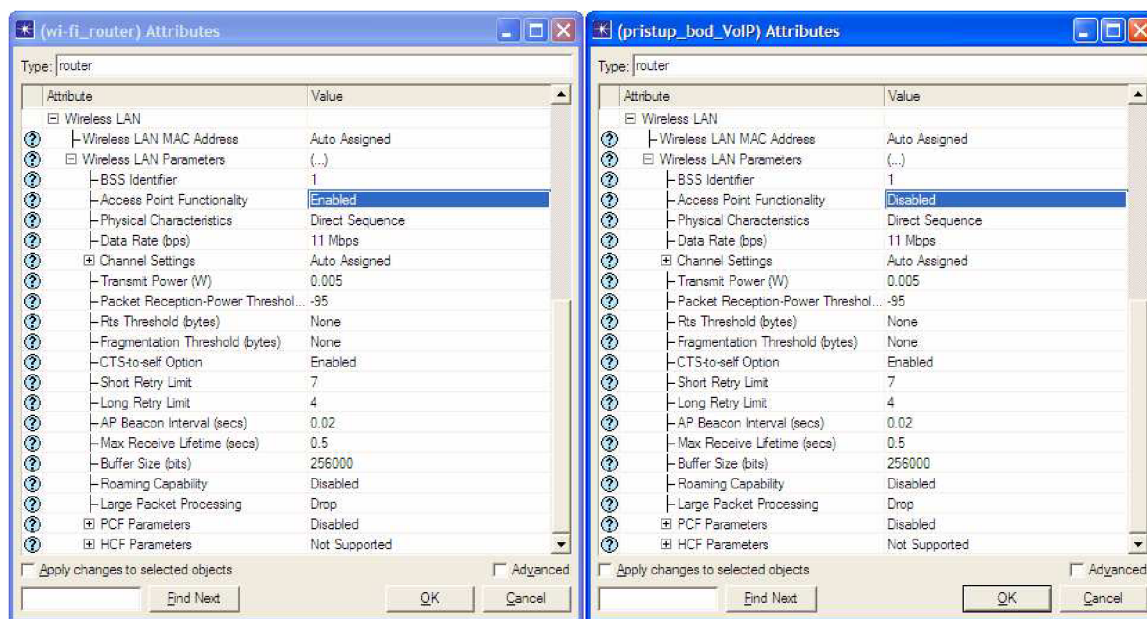


Obr.6.9 Nastavenie atribútov Attributes telefónu VoIP_1_a.

6.3.3 Nastavenie Wi-fi siete

Bezdrôtová sieť sa skladá z wi-fi routra *wi-fi_router*, troch wi-fi klientov *WkstA*, *WkstB* a *WkstC* pripojených do siete cez wi-fi router a nakoniec prístupového bodu *pristup_bod_VoIP*, ktorý slúži ako prístupový bod telefónu *VoIP_3_b*. Na wi-fi routry v položke *Wireless Lan Parameters* nastavíme označenie prístupového bodu (Access Point) na ktorý sa budú wi-fi klienti pripojovať. Toto nastavenie AP definujeme v kolónke *BSS Identifier = 1*, ktoré rovnako nastavíme aj na wi-fi klientoch ako aj na prístupovom bode pre VoIP telefón. Ďalej nastavíme wi-fi router ponuku *Access Point Functionality* a to na

Enabled (povolený) a tak ho nastavíme ako AP tejto siete. **Prístup_bod_VoIP** naopak nastavíme na **Disabled** (zakázaný), čo je z toho dôvodu aby sa správval len ako prístup pre **VoIP_3_b** telefón (viď.Obr.6.10).



Obr.6.10 Nastavenie prístupového bodu Access Point Functionality wi-fi_router.

Nastavenie wi-fi klientov je obdobné ako nastavenie klienta, ktoré sme už popisovali (viď.Kap.6.3.1). Nasledujúca tabuľka zobrazuje služby na jednotlivých klientoch (Tab.2).

Tab.2 Služby na jednotlivých klientoch.

Názov klienta	Služba ftp	Služba http	Služba voip
Wkst_1	Áno	Áno	Áno
Wkst_2	Áno	Nie	Áno
Wkst_3	Áno	Nie	Áno
Wkst_4	Áno	Áno	Áno
Wkst_5	Áno	Áno	Nie
Wkst_6	Áno	Áno	Áno
Wkst_A	Áno	Áno	Nie
Wkst_B	Áno	Nie	Nie
Wkst_C	Nie	Áno	Nie

6.4. Nastavenie testovaných parametrov – štatistik

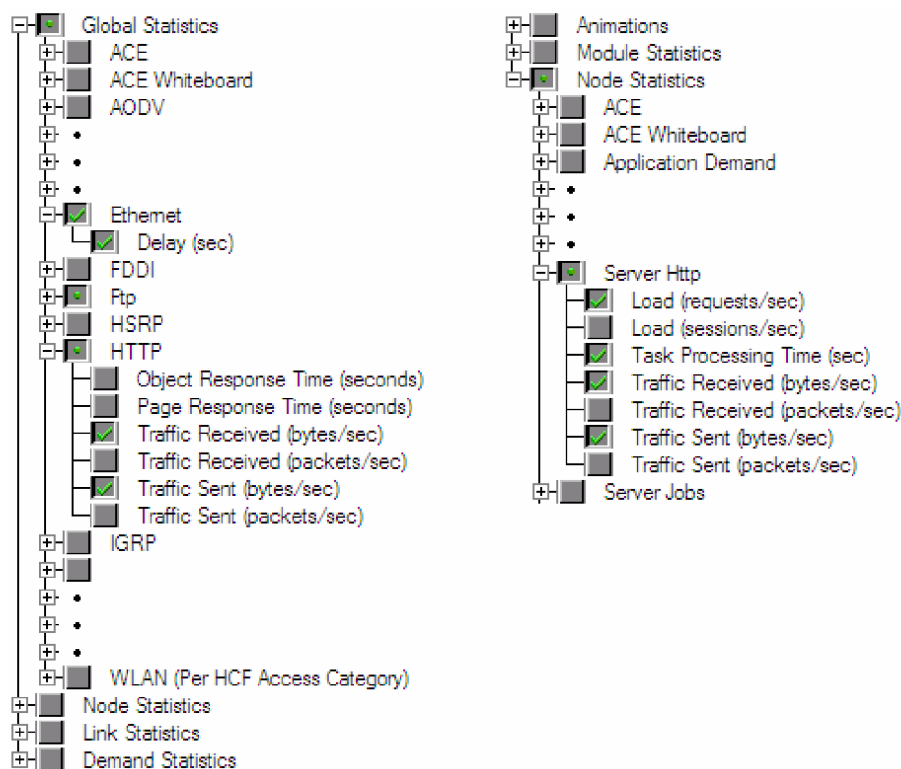
Ide o nastavenie výstupu simulácie, čiže zachytenie nasimulovaných hodnôt. Tieto hodnoty sú zobrazované v grafoch, kde na *ose x* je vynášaný čas a na *ose y* je vynášaná meraná veličina.

Aby sme mohli zobrazovať výsledok štatistiky (určitý graf), musíme najprv nastaviť položku **Choose Individual DES Statistic**, cez ktorú vyberieme jednotlivé sledované parametre [9].

Tieto parametre môžeme nastaviť dvoma spôsobmi:

- **Globálne** – nastavujeme vybrané parametre na každom objekte v sieti.
- **Lokálne** – nastavujeme vybrané parametre len na určitom objekte v sieti.

Príklady nastavení globálnych a lokálnych parametrov zobrazuje obrázok 6.11.

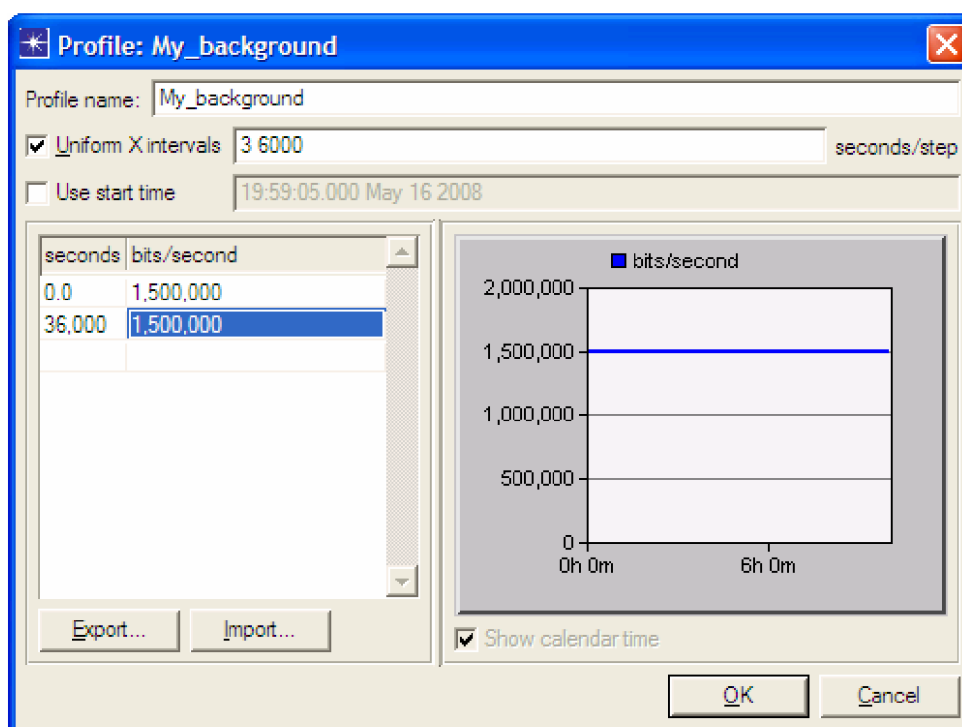


Obr.6.11 Príklad nastavenia globálnych (vľavo) a lokálnych (vpravo) parametrov.

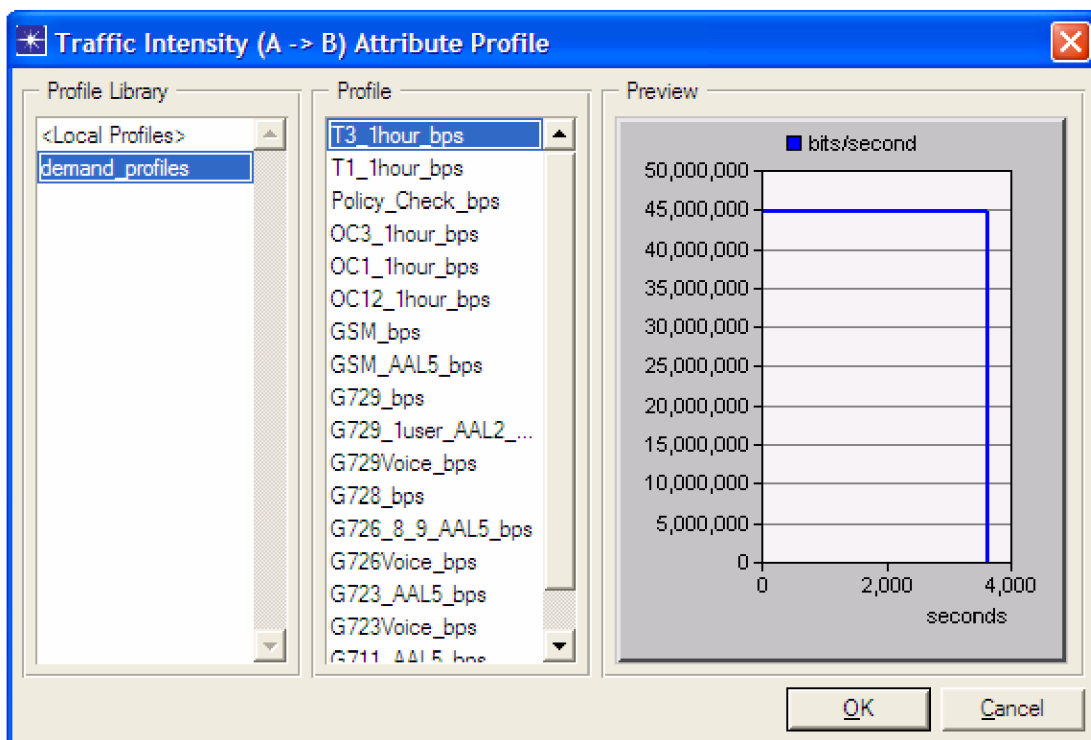
6.5. Nastavenie prevádzky na pozadie

OM nám ponúka možnosť zaťažiť sieť určitou intenzitou prevádzky na pozadie. **Background Load** je atribút každého linku, keďže chceme nastaviť prevádzku na cele pozadie siete a nie jednotlivivo pre každé spojenie uzlov, OM nám ponúka zrýchlený postup nami zamýšľaného nastavenia.

Pravým tlačidlom klikneme na ľubovoľnú linku (napr. *router_2* → *router_3*) a vyberieme **Select Similar Links**, tým označíme všetky podobné linky. Opäť klikneme pravým tlačidlom a z menu vyberieme **Edit Attributes – Background Load – Edit** vyberieme položku **Intensity (bps) [router_2 → router_3]**. Následne máme dve možnosti, buď zvolíme položku **Edit** alebo **Select**. Položkou **Edit** editujeme obrázok 6.12 zobrazuje nakonfigurované hodnoty v pozadí o veľkosti 1500kbit/s na dobu jednej hodiny. Položkou **Select** je možné vybrať buď nami definovanú prevádzku, alebo si vyberieme z už preddefinovaných prevádzok (viď.Obr.6.13).



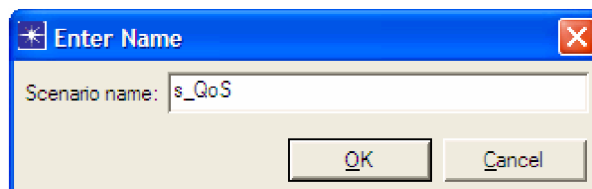
Obr.6.12 Nastavenie vlastnej prevádzky na pozadie.



Obr.6.13 Nastavenie preddefinovanej prevádzky na pozadie.

6.6. Vytvorenie topológie s QoS

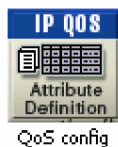
Pri vytváraní topológie siete *s_QoS* si pomôžeme už navrhnutou sieťou *bez_QoS*, preto zvolíme v menu položku *Scenarios – Duplicate Scenario...*, čo nám zaistí skopírovanie aktuálneho scenára a uložíme si ho pod názvom *s_QoS* (viď.Obr.6.14).



Obr.6.14 Vytvorenie nového scenára s_QoS.

Na ploche OM máme už vytvorené objekty *Application Config* a *Profile Config*, zo skopírovaného scenára *bez_QoS*, tieto objekty sú už aj nastavené, čo nám uľahčí prácu

s návrhom siete s *QoS*, keďže ich nastavenia nemusíme znovu nastavovať. Ako ďalší vložíme na plochu objekt *QoS Config* (Obr.6.15).



Obr.6.15 Objekt QoS Config.

Pre konfiguráciu parametrov QoS klikneme pravým tlačidlom na tento objekt a vyberieme položku *Edit Attributes*. OM nám ponúka rôzne mechanizmy pre zaistenie QoS v sieti:

- *CAR Profiles* – využíva sa pre klasifikáciu a riadenie šírky pásma, využíva CoS v hlavičke v IPv4,
- *Custom Queuing Profiles* – užívateľský definovaný spôsob riadenia fronty,
- *FIFO Profiles* – riadenie do front bez priority (prvý prišiel, prvý odíde),
- *MWRR /MDRR /DWRR Profiles* – profily založené na princípu Round-Robin,
- *Priority Queuing Profiles* – riadenie do prioritných front, všetky fronty sú rovnako veľké,
- *RSVP Flow Specification* – špecifikácia premávky podľa RSVP FlowSpec,
- *RSVP Profiles* – riadenie šírky pásma podľa profilov RSVP, využíva sa u technológie IntServ,
- *WFQ Profiles* – riadenie do front podľa priorít.

Pre nasledujúce nastavenie sme zvolili použitie mechanizmu WFQ (viď.Kap.6.5.1), pretože pre návrh modelu diffserv domény je najvhodnejší [9].

Po rozkliknutí ponuky *WFQ Profiles* máme na výber viac profilov, podľa ktorých sa nám budú pakety riadiť do front:

- *ToS* – triedenie do front podľa položky ToS v hlavičke IP paketu v4,
- *Protocol* – triedenie do front podľa použitého protokolu,
- *Port* – triedenie do front podľa použitého portu,
- *DSCP* – triedenie do front podľa hodnôt DSCP.

V našom modeli *s_QoS* značkovanie prebieha na okrajových smerovačoch a priradujú paketom hodnotu DSCP, preto bude vhodné použiť profil *DSCP*.

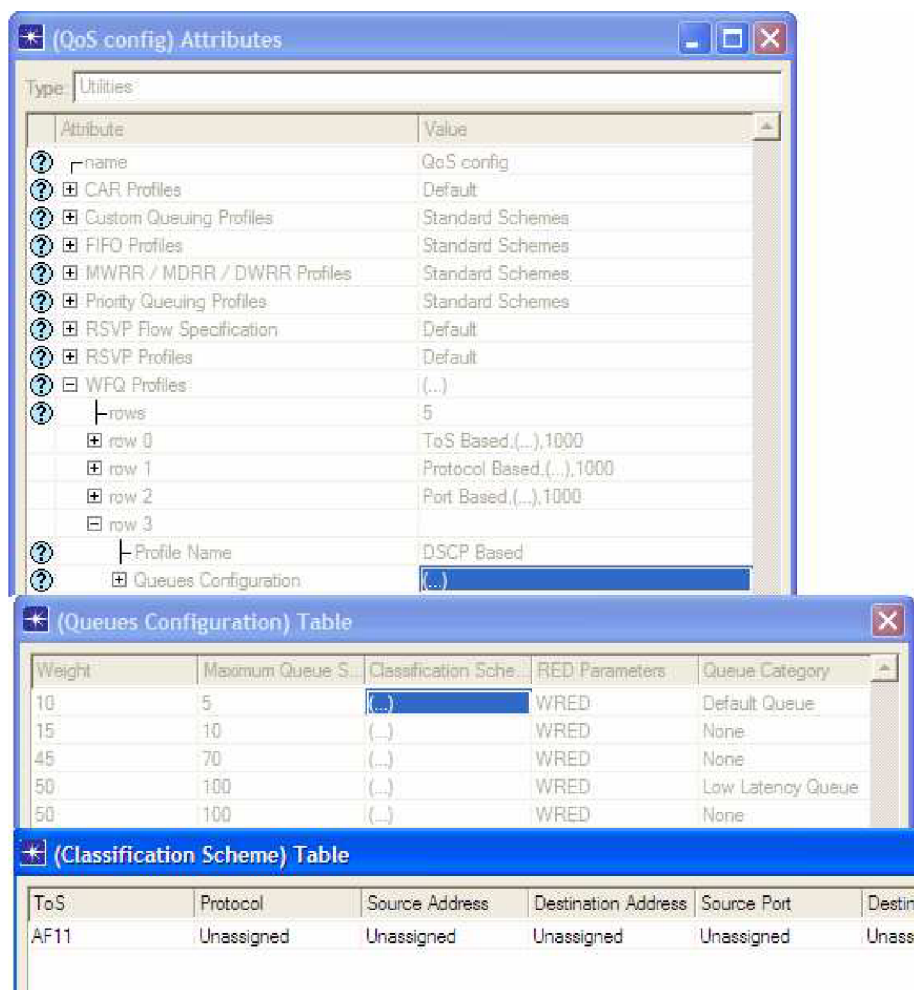
Položka *Queues Configuration* slúži pre nastavenie jednotlivých front. Do položky *rows*, ktorá nám určuje počet front zdefinujeme hodnotu **5**.

Pre ukážku si opäť ukážeme jedno nastavenie podrobnejšie a to položku *row 0*, pre ktorú nastavujeme nasledujúce hodnoty (viď.Tab.3 a Obr.6.15):

- **Weight** – váha fronty (udáva sa pre každú frontu zvlášť). Táto hodnota odpovedá relatívnej veľkosti šírky pásma priradenej danej fronte.
- **Maximum Queue Size [pkts]** – udáva maximálny počet paketov vo fonte.
- **Classification Scheme:**
 - *rows* – položka *rows* udáva počet kritérií (napr. značiek) priradených jednej fronte:
 - **ToS** – zaťaženie do fronty podľa hodnoty ToS alebo DSCP,
 - **Protocol** – zaradenie do fronty podľa protokolu,
 - **Source Address** – zaradenie do fronty podľa cieľovej adresy,
 - **Source Port** – zaradenie do fronty podľa zdrojového portu,
 - **Destination Port** – zaradenie do fronty podľa cieľového portu,
 - **Incoming Interface** – zaradenie do fronty podľa prichádzajúceho rozhrania.
- **RED Parameters** – nastavenie parametrov RED/WRED (viď.Kap.6.5.2):
 - **Red Status,**
 - **Exponential Weight Factor,**
 - **Minimum Threshold,**
 - **Maximum Threshold,**
 - **Mark probability factor,**
 - **CE Marking.**
- **Queue Category** – táto hodnota udáva, či je použitá fronta *None*, *Low Latency Queue* (fronta s nízkym oneskorením), alebo fronta *Default Queue*. Fronta s nízkym oneskorením predstavuje presne dané priority v mechanizme WFQ. Túto a rovnako aj frontu *Default Queue* je možné použiť iba raz. Dátový tok, ktorý vstupuje do fronty je *Low Latency Queue*, keď je táto prázdna, môžu sa odosielať dáta z iných front.

Tab.3 zaradenie front a priradená váha.

<i>Fronta v OM</i>	<i>Podtrieda PHB</i>	<i>Váha (Weight)</i>
<i>Q1-ftp</i>	AF11	10
<i>Q2-http</i>	AF21	15
<i>Q3-voip_1</i>	EF	45
<i>Q4-voip_2</i>	EF	50
<i>Q5-voip_3</i>	EF	50



Obr.6.15 Nastavenie atribútov QoS Config.

6.6.1 Profil WFQ (Weighted fair queueing)

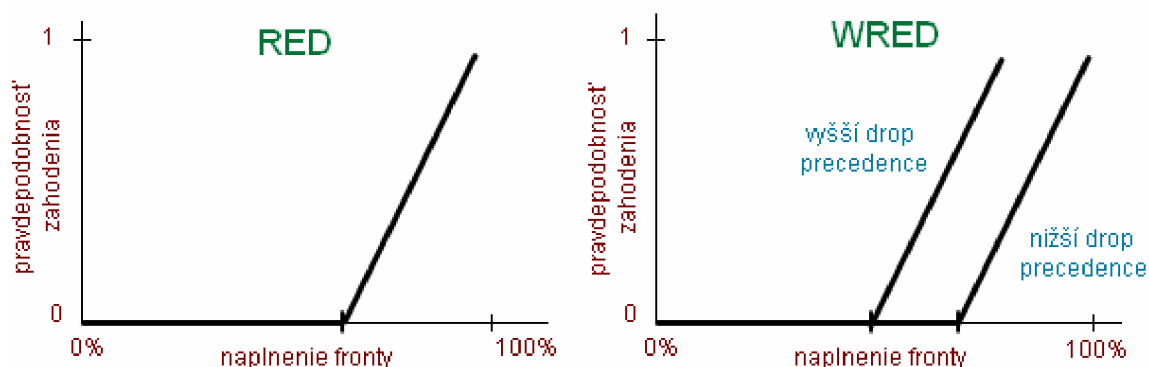
Pri profile WFQ sú postupne obsluhované vždy všetky fronty. Žiadna fronta nemá vyššiu prioritu ako iná fronta. Presný postup pri určovaní, z ktorej fronty má byť vybraný ďalší paket k odoslaniu je nasledovný. Pre každý prichádzajúci paket je vždy vypočítaný čas, kedy najneskôr musí byť daný paket odoslaný vzhľadom k tomu, že pre obsluhu fronty do ktorej je paket zaradený, je garantovaná určitá kapacita linky. Napríklad, ak je paket vkladajú do fronty obsluhovanej rýchlosťou 1 paket za 3 jednotky času (pre jednoduchosť uvažujeme pakety s rovnakou dĺžkou), pričom sú vo fronte už vložené 2 pakety, potom tento nový paket musí byť obslužený najneskôr za 9 jednotiek času od jeho príchodu. Po ukončení obsluhy každého paketu sa vždy určí, ktorý paket zo všetkých paketov na vrchole neprázdnych front má byť obslužený najskôr [16].

6.6.2 Metódy RED/WRED

RED (Random Early Detection) a WRED (Weighted Random Early Detection) sú metódy prevencie zahlcovania (congestion prevention). Pokiaľ necháme fronty úplne naplniť paketmi a až potom začneme zahadzovať ďalšie prichádzajúce pakety, môže dôjsť k synchronizácii zahltenia medzi viacerými smerovačmi a vytvorenie vln striedajúcich zahltenie s okamžikmi, kedy je znížený objem dát posielaných do siete tak, že nie je využitá kapacita liniek. Príčinou tohto javu je chovanie protokolu TCP, ktorým sa dnes významnou časťou prenáša významná časť všetkých dát v sieti Internet. Pokiaľ totiž protokol TCP na strane odosielaťa detekuje stratu paketu, odošle stratený paket znovu a zároveň zníži rýchlosť odosielania paketov k príjemcovi, pretože predpokladá preťaženie prenosovej cesty a znížením rýchlosti chce predísť ďalšej strate paketov. Pretože smerovačom obvykle prechádza rada rôznych TCP spojení súčasne od rôznych odosielačov, dôjde pri zahltení smerovača ku zníženiu rýchlosti na niekoľkých odosielačoch a tým aj k následnému nevyužití kapacity výstupnej linky smerovača. Jednotliví odosielaelia postupne obnovia pôvodnú rýchlosť prenosu a dôjde k zahlteniu smerovača opäť.

Tomuto javu sa snaží predísť metóda RED (viď.Obr.6.16). Ak presiahne naplnenie fronty určitý limit, začne smerovač zahadzovať pakety z náhodne vybraných TCP spojení. Pravdepodobnosť zahodenia paketu sa zvyšuje so zvyšujúcim sa naplnením fronty. Tým dôjde ku zníženiu objemu dát od niektorých odosielačov a plynulému vyrovnaniu celkového objemu prichádzajúcich dát s kapacitou odchádzajúcej linky.

Pri metóde WRED (viď.Obr.6.16) závisí pravdepodobnosť zahodenia paketu aj na značke paketu pridelenéj klasifikáciou. Limit zahodenia paketu pri prekročení naplnenia fronty, môže byť rôzny pre pakety s rôznymi značkami [16].

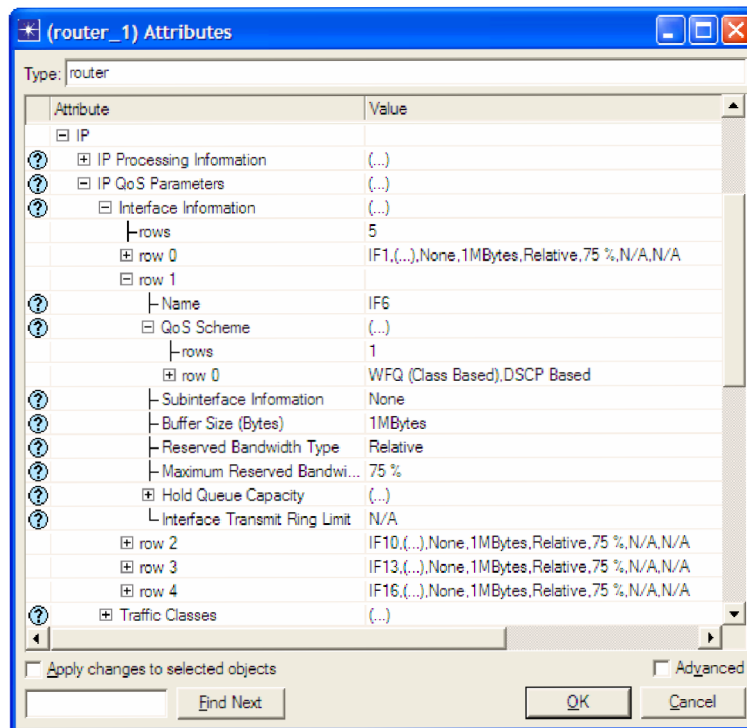


Obr.6.16 Metóda RED (vľavo) a metóda WRED (vpravo) [16].

6.6.3 Nastavenie kvality služieb na smerovači

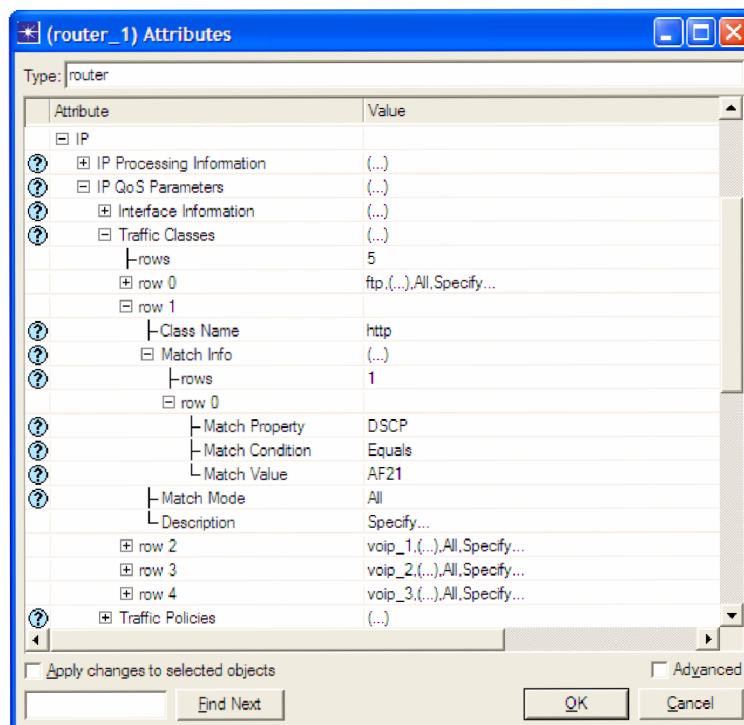
Pre správnu funkciu podpory QoS je nutné, aby na smerovači, ktorý prevádza klasifikáciu, alebo riadenie do front bolo nastavené *QoS Schema*. Keďže fronty už máme nastavené bude vysvetlený iba postup ako nastaviť smerovač pre podporu už vytvorených front.

V menu zvolíme *Edit Attributes – IP – IP QoS Parameters* potrebujeme na určitý port smerovača nakonfigurovať nami vytvorené QoS schéma (WFQ Profil). To prevedieme kliknutím na položku *Interface Information* a nastavíme hodnoty podľa nasledujúceho obrázku (viď.Obr.6.17).

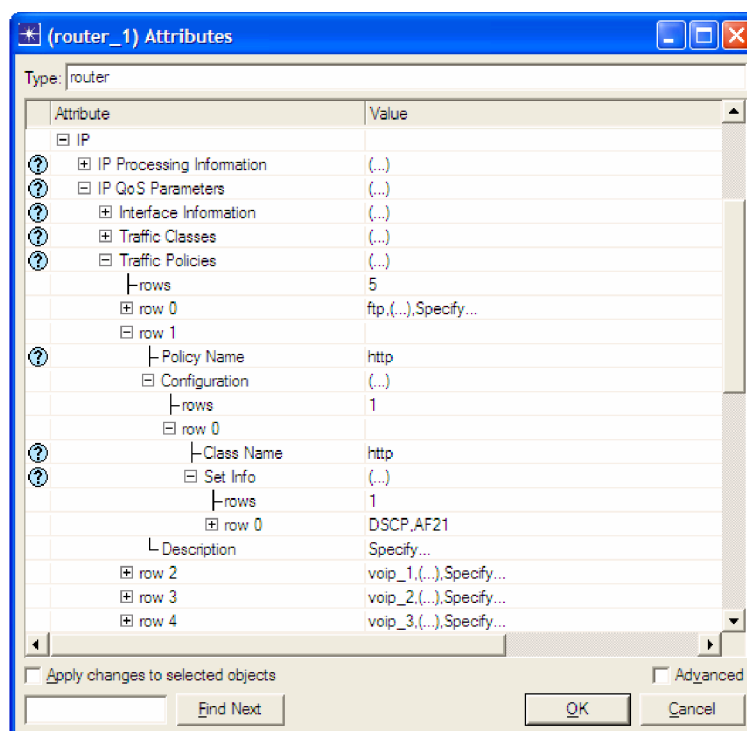


Obr.6.17 Nastavenie atribútov Iterface Information na smerovači.

Ďalej môžeme v OM konfigurovať parametre QoS cez položku *Traffic Classes* (Obr.6.18) a *Traffic Policies* (Obr.6.19).



Obr.6.18 Nastavenie atribútov Traffic Classes na smerovači.

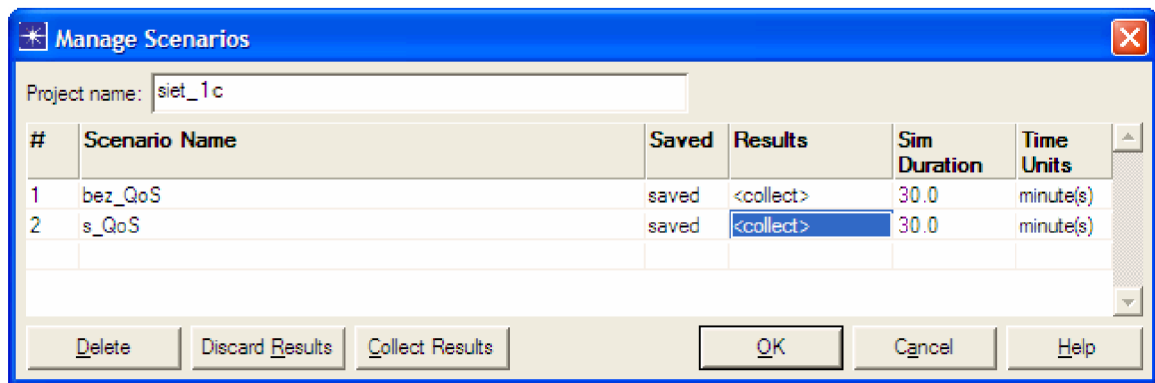


Obr.6.19 Nastavenie atribútov Traffic Policies na smerovači.

6.6. Nastavenie simulácie

Keďže nás zaujíma porovnanie nastavených parametrov siete *bez_QoS* a *s_QoS* (viď.Kap.6.4 a Tab.2), využijeme ďalšie zjednodušenie, ktoré nám OM ponúka a to nastavenie simulácií pre oba scenáre súčasne. Simulujeme tak scenáre súčasne a následne po simulácií je možné zobraziť charakteristiku určitého uzlu oboch scenárov do jedného grafu, čo sa nám bude hodiť.

Nastavenie prevedieme nasledovne. Z menu v položke *Scenarios* vyberieme *Manage Scenarios* (viď.Obr.6.20), čo nám umožní spravovať všetky vytvorené scenáre, prípadne ak niektorý zo scenárov chceme vymazať, môžeme tak spraviť kliknutím na tlačidlo *Delete*. Pre oba scenáre zvolíme v položke *Results* hodnotu *collect*. Položka *Sim Duration* určuje dobu, ktorú chceme odsimulovať. V našom prípade nastavíme hodnotu **30 minutes**. Stlačením tlačidla *OK* spustíme testovanie oboch scenárov súčasne.



Obr.6.20 Nastavenie atribútov oboch scenárov pre simuláciu.

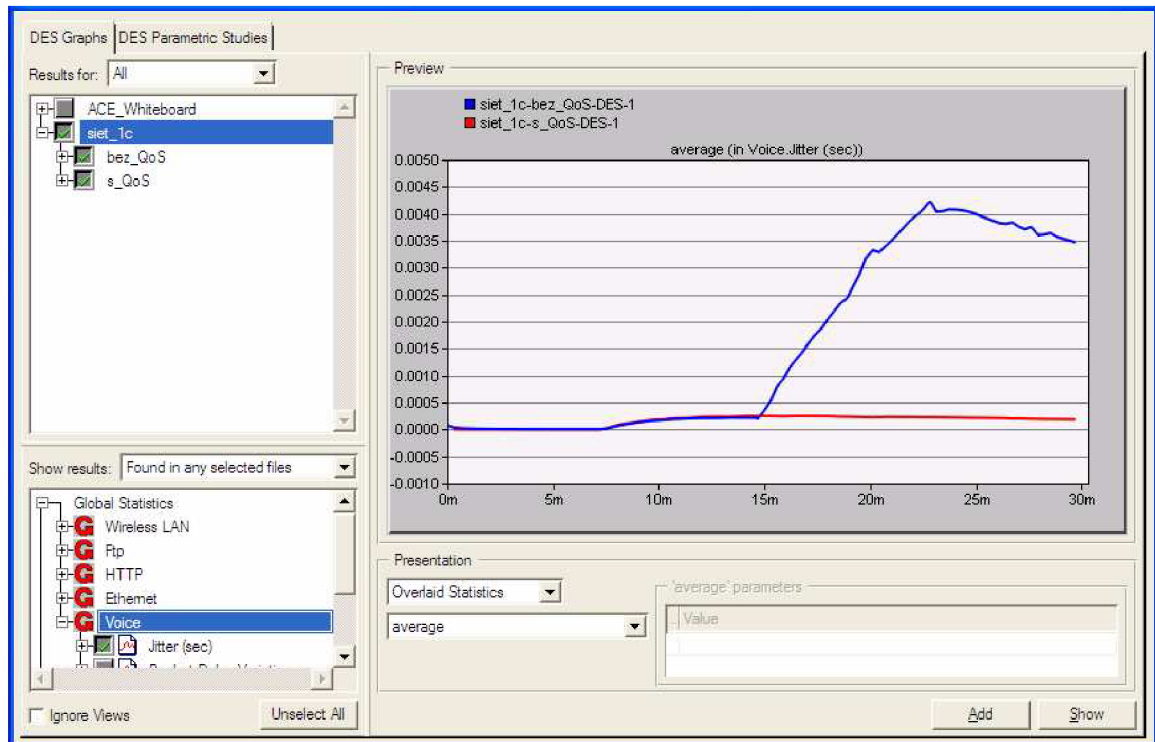
6.7. Zobrazenie výsledkov simulácie

Keďže sme nastavili rovnaké štatistiky pre oba scenáre, budeme porovnávať namerané hodnoty z oboch scenárov zároveň.

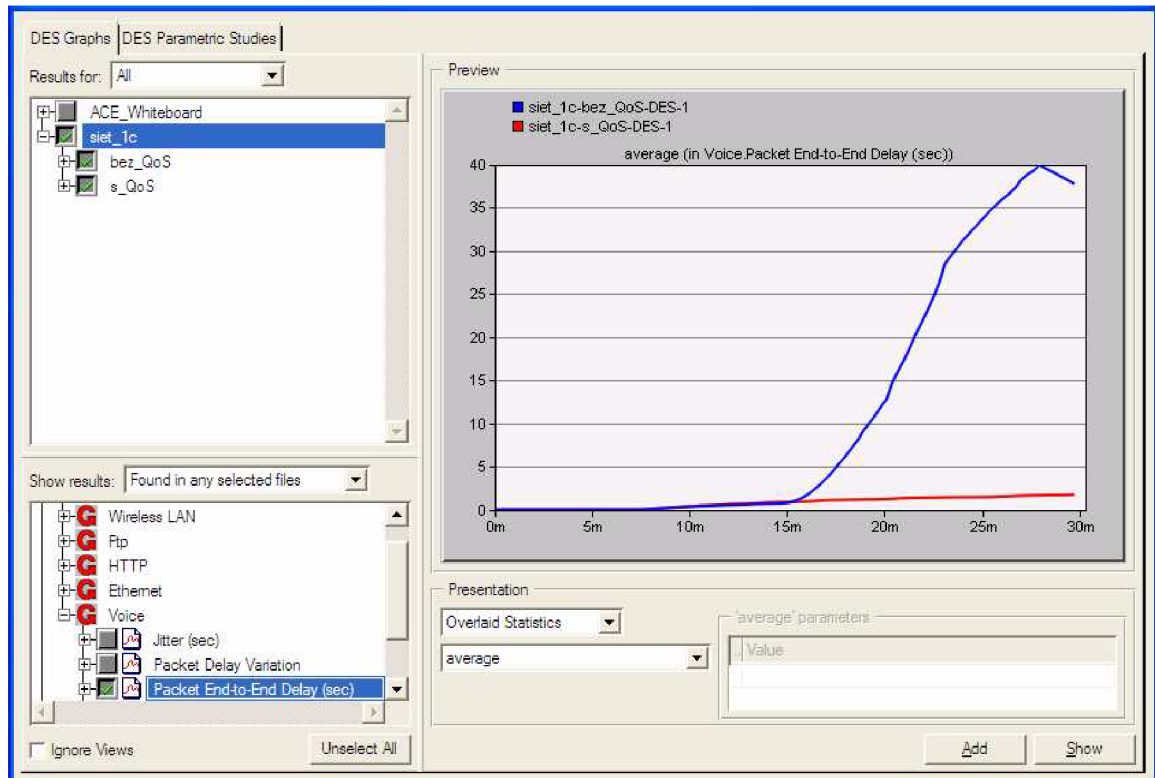
Klikneme pravým tlačidlom myši na plochu a zvolíme položku **View Results**. V novom okne zmeníme hodnotu položky **Result for** na **Current Project**. Označíme zobrazenie pre oba scenáre a následne rozvetvíme štatistiku **Object Statistics – voip_3_b – jitter -->**. Dole pod grafom zmeníme hodnotu zo **Stacked Statistics** na **Overlaid Statistic** a rovnako aj **As is** na **average** (grafy sa nám spriemerujú).

Globálne štatistiky sa zobrazujú rovnakým postupom ako objektové, jediný rozdiel je pri rozvetvení, kedy zvolíme položku **Global Statistics** a nasledovne pokračujeme už popisovaným postupom až sa dostaneme k nasledujúcemu zobrazeniu (viď.Obr.6.21 a Obr.6.22).

Kliknutím na tlačidlo **Show** sa nám zobrazia podrobnejšie charakteristiky. Kliknutím pravým tlačidlom na graf z ponuky vyberieme **Export data to Spreadsheet**, nám OM vyexportuje hodnoty do súboru (napr.*siet_1c-be2.txt*). Z týchto hodnôt a použitím programu na tvorbu grafov sme zostavili vybrané charakteristiky, ktoré nájdete na konci práce (viď. Príloha 1-6).



Obr.6.21 Zobrazenie globálnej štatistiky voice, zmena omeškania (jitter).



Obr.6.22 Zobrazenie globálnej štatistiky voice, oneskorenie koncových uzlov (delay) .

6.7. Výsledky simulácie

Simuláciou prevádzok sme si overili správne nastavenie QoS pomocou profilu WFQ a triedenie do front podľa hodnôt DSCP. Z toho môžeme usudzovať, že klasifikácie a značenie paketov dátových tokov jednotlivých klientov a nastavenie ich smerovania na smerovačoch funguje bez chýb. Taktiež z toho plynie, že vybrané nastavenie sa ukázalo ako správna voľba pre zabezpečenie kvality služieb pre technológiu DiffServ. Z výsledkov rovnako vidno akú úlohu hrá počet smerovačov medzi uzlami, ktoré medzi sebou komunikujú. V prílohe č.2, kde je zobrazené časové chvenie klienta VoIP_2_a nie je rozdiel až tak rozdielny, naopak je tomu pri zobrazení časového chvenia u klienta VoIP_3_b (príloha č.3), kde červená charakteristika zobrazuje chvenie so zabezpečením kvality služby a ako vidno pakety sa pohybujú konštantne okolo osy s nulovou hodnotou.

V prílohe č.1 môžeme vidieť zobrazené časové chvenie jednotlivých paketov počas simulácie na VoIP servery a v prílohe č.5, ktorá zobrazuje oneskorenie jednotlivých paketov počas simulácie na bezdrôtovej sieti (Wireless).

Záver

Cieľom tejto práce bolo preštudovať používané technológie pre zaistenie kvality služieb v paketových sieťach. Práca obsahuje popis historického vývoja QoS, zhrnutie výhod a nutných bodov pre zavedenie QoS. V tab.1 je zobrazená citlivosť rôznych typov služieb na parametre siete a je tak zreteľne vidno ktorá služba je na jednotlivé parametre siete náchylná viac a ktorá menej.

V kapitole, ktorá nesie názov „Sieťová vrstva – Protokoly IP“ sú rozdelené modely QoS, ktoré sú podrobne prebrané v ďalších kapitolách a to predovšetkým modely IntServ a DiffServ. Každý z modelov pristupuje k riadeniu kvality služieb iným spôsobom, no najlepšie výsledky sú dosiahnuté ich kombináciou, ktorú zobrazuje obr.7.

Praktická časť obsahuje návrh, nastavenie a samotnú simuláciu siete mechanizmu QoS, realizovanej pomocou vybraného modelu DiffServ. Navrhnutá sieť obsahuje dva scenáre, jeden bez QoS a druhý so zaistením kvality služieb pomocou profilu WFQ a triedenia do front podľa hodnôt DSCP.

Výsledkom simulácie sme je potvrdenie predpokladu, že navrhnutá sieť bez QoS nebude dostatočujúca pri väčšom zaťažení siete a naopak sieť so zabezpečením kvality služieb je tou správnou voľbou pre dosiahnutie čo najlepších výsledkov pre nami navrhnutú sieť.

Zoznam literatúry a použitých zdrojov

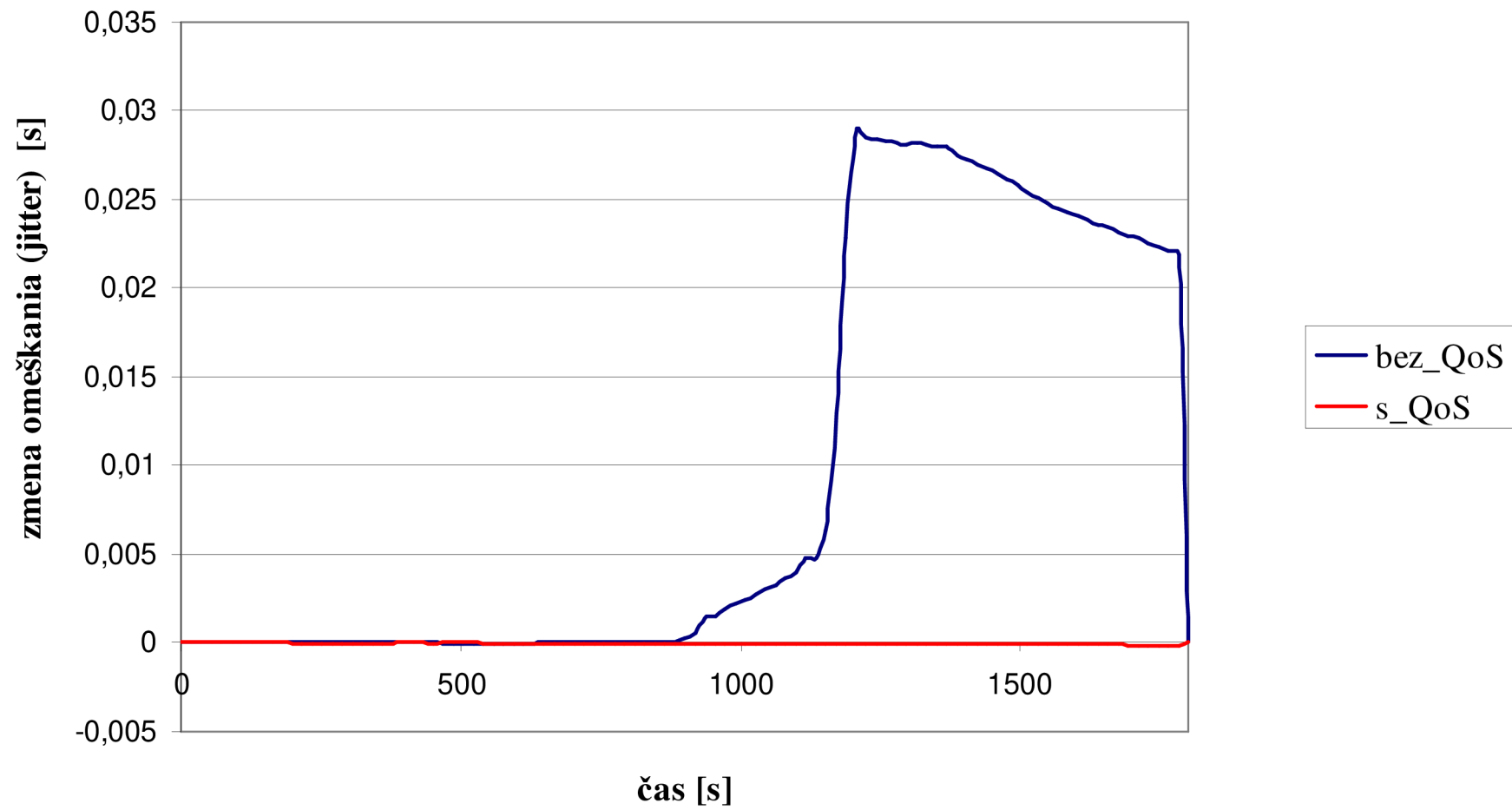
- [1] PARK–Kun, I. *QoS in Packet Networks*. The MITRE Corporation USA, 2005, ISBN: 0-387-23390-3
- [2] ODVÁRKA, Petr. *Řízení šířky pásma v LAN*. Svět sítí & Infinity a.s. 2001. [cit. 2008-05-07]. Dostupné z:
< http://www.svetsiti.cz/view_list.asp?rubrika=Tutorialy&temaID=62 >.
- [3] *Streaming*. Visual Connection. a.s. 2006. [cit. 2008-05-07]. Dostupné z:
< <http://www.visual.cz/cs/site/produkty/Technologie/Streaming.htm> >.
- [4] PANÁČEK, Petr. *QoS v IP sítích*. ANECT a.s. 2002 [cit. 2008-05-07]. Dostupné z:
< <http://www.anect.com/cs/info/tiskove-centrum/clanky/qos-v-ip-sitich.html> >.
- [5] GIERLI, Tomáš. *QoS Hodnotenie kvality služieb v systémoch prenosu údajov*. 2006 [cit. 2008-05-07]. Dostupné z:
< http://www.qos-diplomka.webzdarma.cz/4_QoS_v_IP/442a.HTM >.
- [6] KACÁLEK, Jan. *QoS v počítačových sítích*. 2006 [cit. 2008-05-07]. Dostupné z:
< <http://amarok.cesketelekomunikace.cz/xkacal00/?action=rout> >.
- [7] BEZCHLEBA, J. *Modelování řízení provozu v hraničních směrovačích*. VUT Brno, 2006. 47 s. Vedúci bakalárskej práce Molnár Karol.
- [8] TIRINDA, V. *Klasifikace a značkování v hraničních směrovačích DiffServ domény*. VUT Brno, 2006. 49 s. Vedúci bakalárskej práce Molnár Karol.
- [9] ZEMAN, O. *Modelování chování páteřních směrovačů DiffServ domény*. VUT Brno, 2006. 71 s. Vedúci bakalárskej práce Molnár Karol.
- [10] TOMAN, P. *Modelování zajištění kvality služeb v síti UMTS*. VUT Brno, 2006. 57 s. Vedúci bakalárskej práce Molnár Karol.

- [11] LATISLAV, R. *Zajištění kvality služeb nad TCP protokolem*. VUT Brno, 2006. 54 s. Vedúci bakalárskej práce Cvrček Daniel.
- [12] DEBRA, Shinder. *Počítačové sítě*. Cisco Systems Inc. 2003, ISBN: 80-86497-55-0
- [13] SZIGETI, Tim. *End-to-End QoS Network Design*. Cisco Systems Inc. 2004. ISBN: 1-58705-176-1
- [14] MOLNÁR, K. – KOVÁŘ, P. – NOVOTNÝ, V. *Současnost a budoucnost VoIP cíť*. VUT Brno, 2007 [cit. 2008-05-07]. Dostupné z: < <http://www.elektrorevue.cz/cz/clanky/komunikacni-technologie/0/soucasnost-a-budoucnost-voip-siti/> >.
- [15] MOLNÁR, K. – ZEMAN, O. *Moderní síťové technologie*. Brno: VUT, Fakulta elektrotechniky, Ústav telekomunikací, 2006 [cit. 2008-05-07]. Elektronické skripta. Dostupné z: < <http://www.utko.feec.vutbr.cz/~molnar> >
- [16] UBIK S. *QoS a diffserv – Úvod do problematiky, 2000* [cit. 2008-05-07]. Dostupné z: < <http://staff.cesnet.cz/~ubik/publications/2000/diffserv.html> >

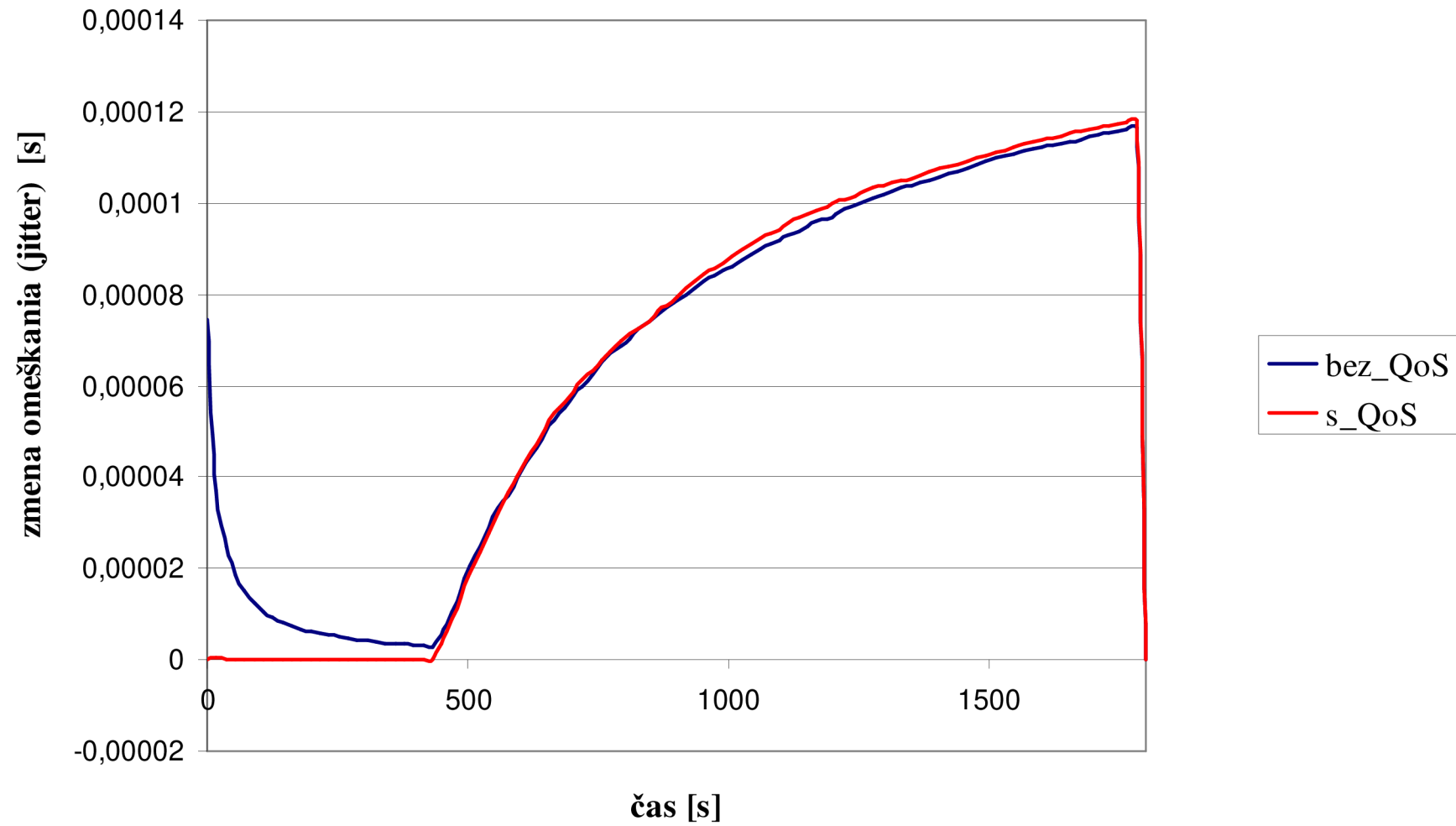
Zoznam príloh

- Príloha č.1 Časové chvenie jednotlivých paketov počas prenosu voip servera
- Príloha č.2 Časové chvenie jednotlivých paketov počas prenosu prenosu na klientovi VoIP_2_a
- Príloha č.3 Časové chvenie jednotlivých paketov počas prenosu na klientovi VoIP_3_b
- Príloha č.4 Oneskorenie jednotlivých paketov počas prenosu na klientovi VoIP_3_b
- Príloha č.5 Oneskorenie jednotlivých paketov počas prenosu na bezdrôtovej sieti (Wireless)
- Príloha č.6 Počet prijatých bytov počas prenosu na klientovi wkstB

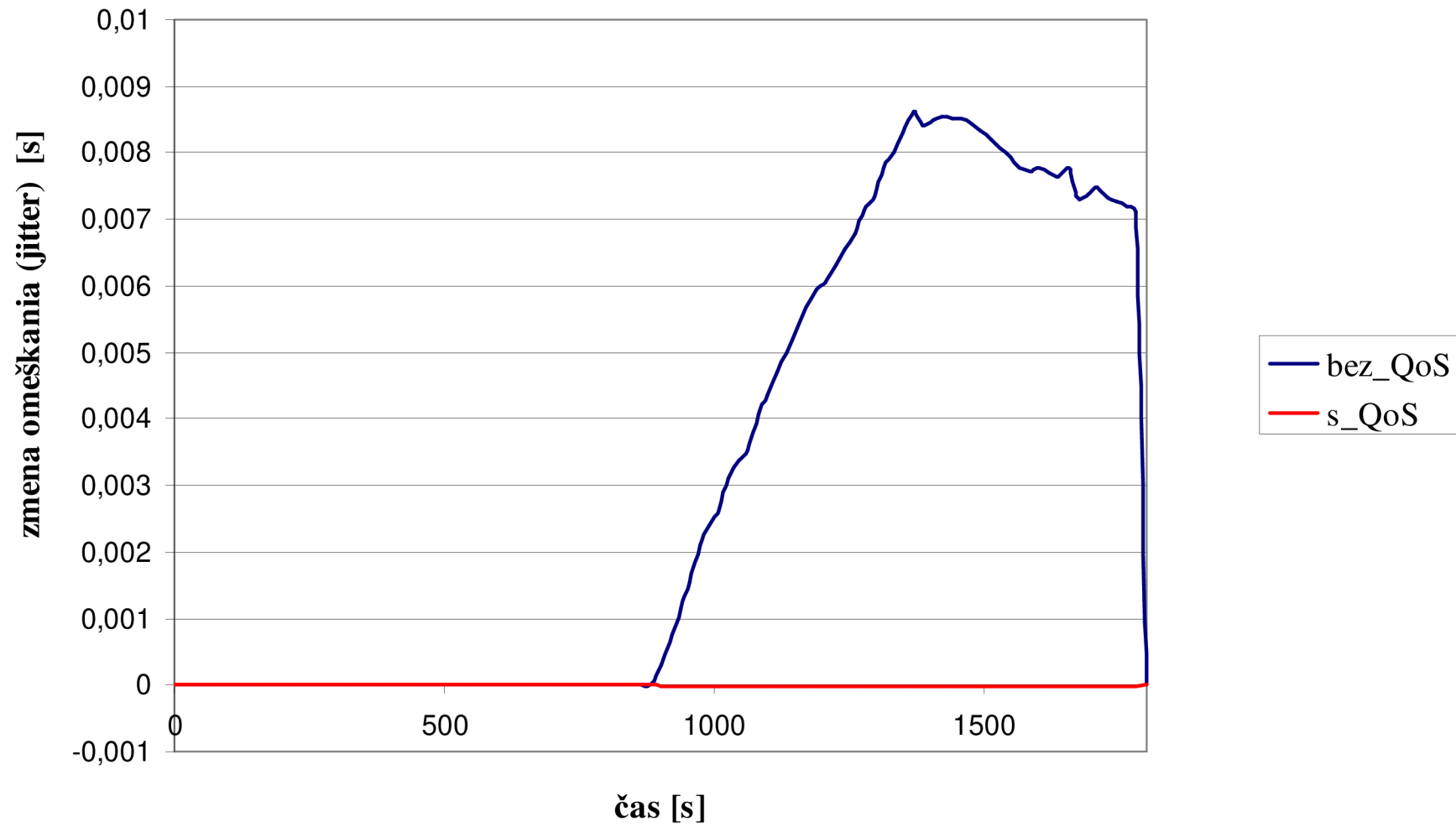
Príloha č.1: Časové chvenie jednotlivých paketov počas prenosu voip servera



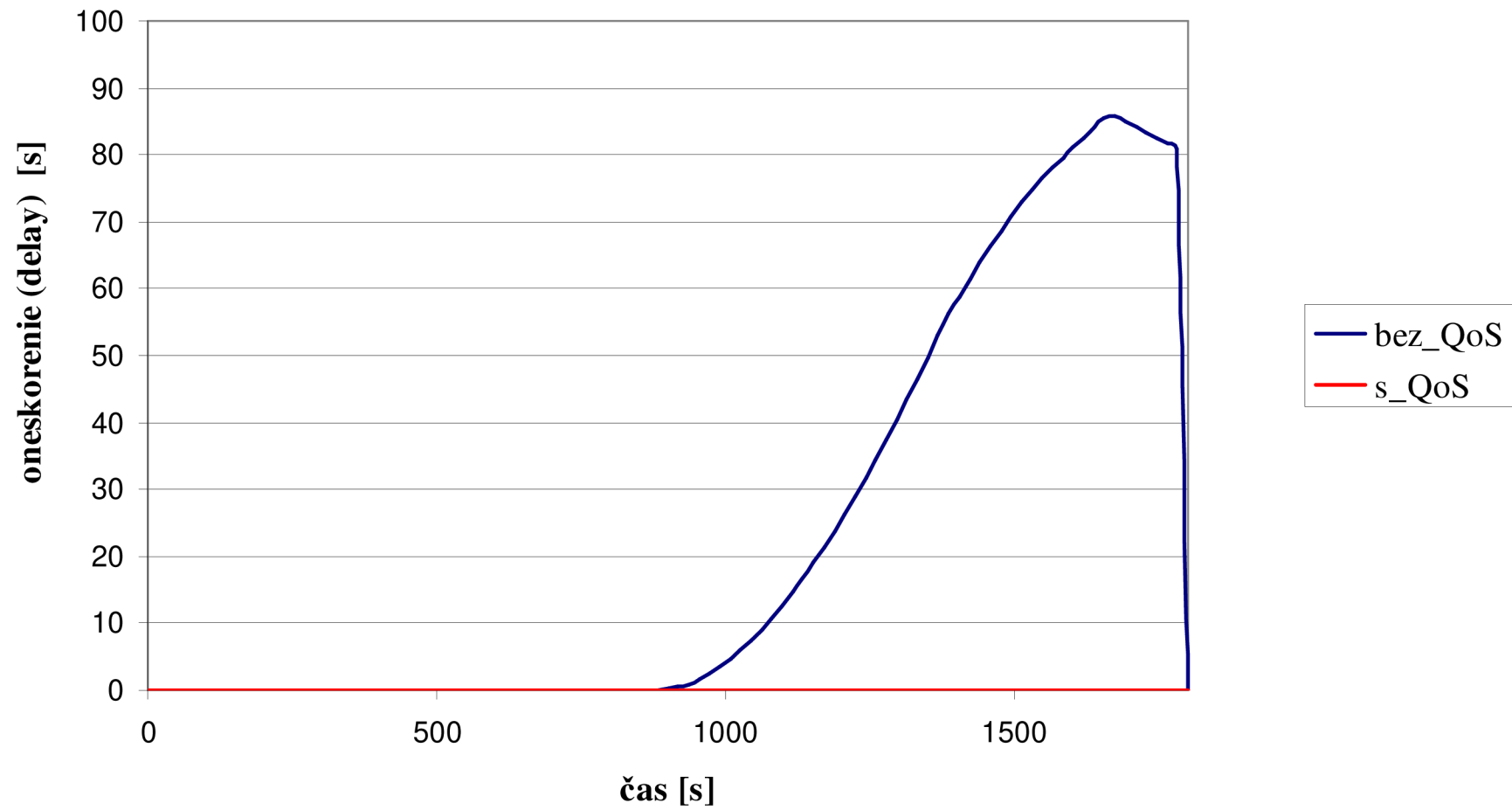
Príloha č.2: Časové chvenie jednotlivých paketov počas prenosu na klientovi VoIP_2_a



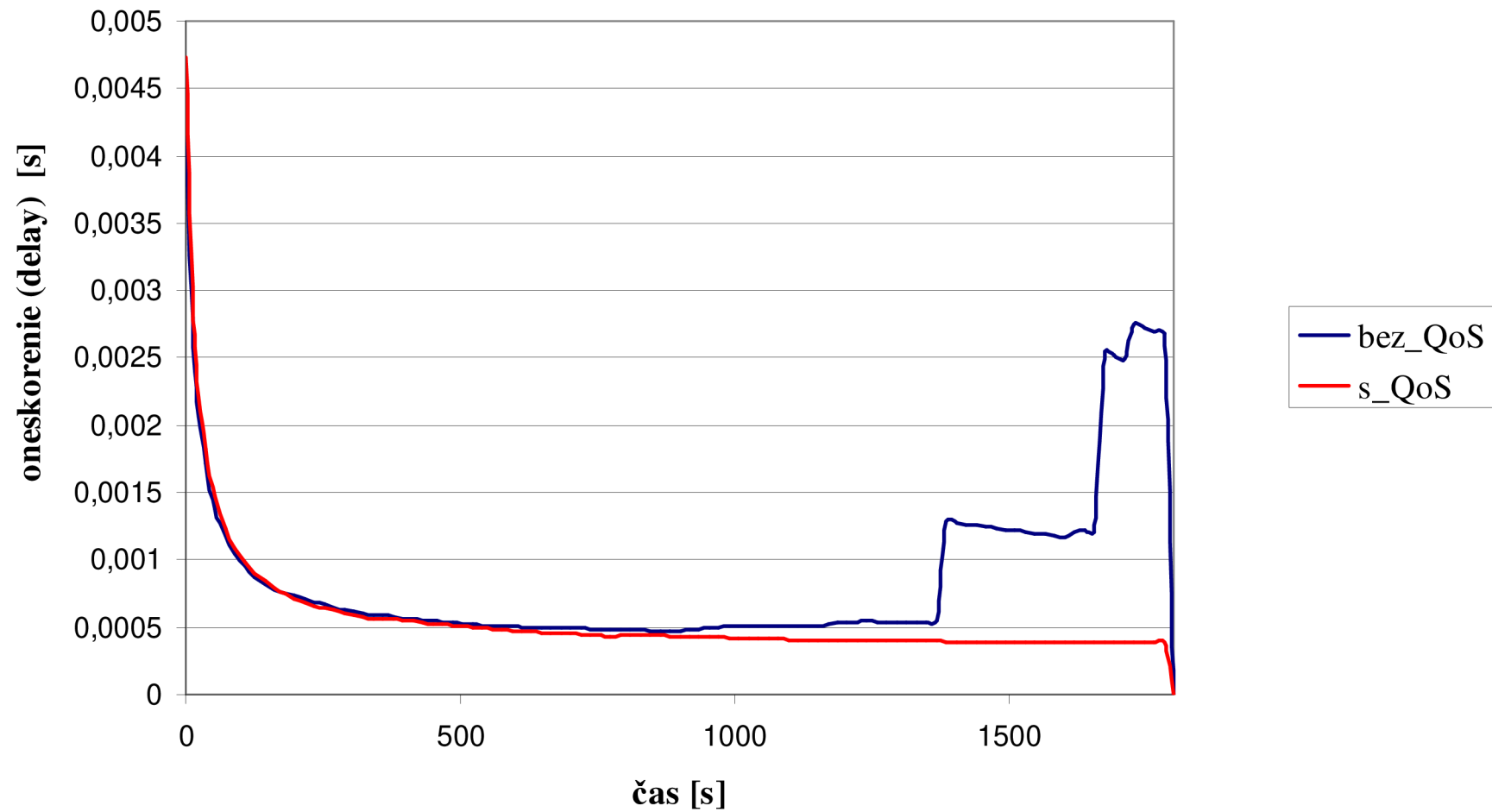
Príloha č.3: Časové chvenie jednotlivých paketov počas prenosu na klientovi VoIP_3_b



**Príloha č.4: Oneskorenie jednotlivých paketov počas prenosu na klientovi
VoIP_3_b**



Príloha č.5: Oneskorenie jednotlivých paketov počas prenosu na bezdrôtovej sieti (Wireless)



Príloha č.6: Počet prijatých bytov počas prenosu na klientovi wkstB

