UNIVERSIDADE FEDERAL DO MARANHÃO
CENTRO DE CIÊNCIAS EXATAS E TECNOLÓGICAS
CURSO DE PÓS-GRADUAÇÃO EM ENGENHARIA DE ELETRICIDADE
ÁREA: CIÊNCIA DA COMPUTAÇÃO

José Coelho de Melo

Estudo da Utilização de Mecanismos de QoS em Redes com Enlaces de Banda Estreita

São Luís - MA
2005

UNIVERSIDADE FEDERAL DO MARANHÃO
CENTRO DE CIÊNCIAS EXATAS E TECNOLÓGICAS
CURSO DE PÓS-GRADUAÇÃO EM ENGENHARIA DE ELETRICIDADE
CIÊNCIA DA COMPUTAÇÃO

Estudo da Utilização de Mecanismos de QoS em Redes com Enlaces de Banda Estreita

Dissertação submetida à Universidade Federal do Maranhão, como parte dos requisitos para a obtenção do grau de mestre em Ciência da Computação.

José Coelho de Melo

Prof. Dr. Zair Abdelouahab
Orientador

São Luís, Maio de 2005

Estudo da Utilização de Mecanismos de QoS em Redes com Enlaces de Banda Estreita

José Coelho de Melo

Esta Dissertação, por meio da comissão julgadora dos trabalhos de defesa de Dissertação de Mestrado, foi julgada adequada para a obtenção do título de "Mestre em Engenharia Elétrica, Área de Concentração Ciência da Computação", e aprovada na sua forma final pelo Programa de Pós-Graduação em Engenharia Elétrica da Universidade Federal do Maranhão, em sessão pública realizada em 14 /03/2005.

Presidente

_____

Prof. Dr. Zair Abdelouahab
(Orientador)

1o Examinador

_____

Prof. Dr. José Neuman Sousa

2o Examinador

_____

Prof. Dr. Francisco José Silva e Silva

Dedicatória

Dedico este trabalho a meus pais, Antônio e Zilmar, que sempre mostraram o valor do conhecimento e se empenharam para que eu tivesse condições de adquiri-lo. A André Lucas e Ana Kelly, que são os principais motivos desta caminhada.

Agradecimentos

A meu orientador, Prof. Zair Abdelouahab, pelas orientações prestadas e pela confiança demonstrada em meu trabalho.

Ao Prof. Francisco José Silva e Silva, que me conduziu como professor e amigo a esta Universidade, e me ajudou muito nos primeiros trabalhos. Sempre serei grato aos olhos que me deu para enxergar mais distante e a acreditar no estudo, na dedicação e na ética.

A DATAPREV, que me acolheu em seu programa de incentivo e me concedeu tempo para frequentar as aulas e para realizar os estudos necessários a este trabalho, além de fornecer os equipamentos para a realização das experiências em laboratório. Agradeço, particularmente, aos gestores João Antônio, Myrna Rios, Margarida Moreira e Cláudia Archer pelas facilidades oferecidas para que eu concluísse este trabalho. A Nilsa que, muitas vezes, me substituiu na realização de tarefas e pelos conhecimentos passados.

A Ângela, minha esposa, que com sua visão crítica me orientou no sentido de otimizar os recursos disponíveis, que pela tolerância às ausências me permitiu conduzir o trabalho com tranquilidade.

E, especialmente, a minha mãe Zilmar, meu pai Antônio e meus irmãos que sempre me apoiaram na buscar novos caminhos.

E a Deus que nos colocou juntos nesta caminhada.

Resumo

Este trabalho apresenta diversas maneiras de utilização dos mecanismos de Qualidade de Serviço (QoS) em redes ligadas por enlaces de banda estreita. É demonstrado que o uso de mecanismos de QoS traz ganhos de performance aos aplicativos prioritários e racionalidade no uso de recursos de banda. O trabalho é desenvolvido em uma plataforma de teste que retrata um seguimento da rede da Previdência Social. São testadas possíveis implementações dos PHBs EF, Afs e BE. São avaliadas as capacidades de oferecer largura de banda garantida, priorização de tráfego, condicionamento de tráfego e o impacto da concorrência entre classes de tráfego.

Os experimentos são montados utilizando a arquitetura de serviços diferenciados (*DiffServ*) da plataforma Cisco, que é o padrão na rede da Previdência Social, e são manipulados os mecanismos de classificação, controle de congestionamento, policiamento e conformação de tráfego. O resultado apresentado dá subsídio para implementação de vários modelos de QoS em redes IP, principalmente para redes servidas por banda estreita.

Palavras-chaves: Serviços Diferenciados (*DiffServ*), Qualidade de Serviço (QoS), redes IP, Política de QoS, Enfileiramento.

Abstract

This work presents several ways of using mechanisms of Quality of Service (QoS) in networks hardwired for links of short band. It is demonstrated that the use of mechanisms of QoS brings profits of performance for priority applications and rationality in the use of band resources. The work is developed in a test platform of Previdência Social. Implementations of the PHBs EF, Afs and BE are tested. The work has evaluated the capacities to offer guaranteed band, traffic priority, conditioning of traffic and the impact of the competition between traffic classes.

The experiments are mounted using the architecture of differentiated services (DiffServ) of platform Cisco, that is the standard in the network of the Previdência Social and are manipulated the mechanisms of classification, control of congestion, policing and conformation of traffic. The results give subsidy for implementation of some models of QoS in IP networks, mainly for those that serve short band.

Keywords: Differentiated Services, Quality of Service, IP Networks, QoS Policy, Queuing.

SUMÁRIO

LISTA DE FIGURAS

LISTA DE TABELA

Lista de Abreviaturas

AF	Assured Forwarding
ATM	Asynchronous Transfer Mode
BA	Behavior Aggregate
BB	Bandwidth Broker
BE	Best Effort
CAR	Committed Access Rate
CBQ	Class-Based Queuing
CBR	Constant Bit Rate
CBWFQ	Class Based Weighted Fair Queuing
COPS	Commom Open Policy Service
CIR	Committed Information Rate
DiffServ	Differentiated Services
DS	Differentiated Services
DSCP	DiffServ Code Point
EF	Expedited Forwarding
FIFO	First In First Out
FTP	File Transfer Protocol
GPS	Global Positioning System
GRED	Generic Random Early Detection
HTTP	Hipertext Transfer Protocol
IETF	Internet Engineering Task Force
IntServ	Integrated Services
IP	Internet Protocol
IPDV	Instantaneous Packet Delay Variation
IPPM	IP Performance Metrics Working Group
ISO	International Standardization Organization
ISP	Internet Service Provider
ITU-T	International Telecommunications Union - Telecommunication Standardization Sector
LAN	Local Area Network
MF	Multi-field

MPEG	Moving Picture Experts Group
MTU	Maximum Transfer Unit
NFS	Network File System
NTP	Network Time Protocol
PHB	Per-Hop Behavior
PQ	Priority Queuing
QDISC	Queue Discipline
QoS	Quality of Service
RED	Random Early Detection
RFC	Request for Comments
RSVP	Resource Reservation Protocol
RTP	Real Time Transport Protocol
RTT	Round Trip Time
SFQ	Stochastic Fairness Queuing
SLA	Service Level Agreement
SLS	Service Level Specifications
TC	Traffic Class
TCP	Transmission Control Protocol
ToS	Type of Service
UDP	User Datagram Protocol
VoIP	Voice over IP
WAN	Wide Area Network
WFQ	Weighted Fair Queuing

1 INTRODUÇÃO

A evolução das redes de computadores trouxe novas funcionalidades, as redes tornaram-se cada vez mais essenciais ao sucesso das corporações e continuam em fase evolutiva. A integração e convergência dos meios de comunicações tornaram as redes de computadores uma infra-estrutura fundamental para que este sucesso aconteça.

Depois de conduzir a todas as partes das corporações os dados contidos nos seus sistemas de informação, a nova realidade das redes de comunicação é a integração e convergência. A partir de agora as redes transportam todo tipo de informação, desde a sinalização de sensores até o tráfego de voz e imagens, tornando-se infra-estrutura indispensável para o funcionamento de todos os serviços de Tecnologia da Informação [65]. Para dar suporte a estes serviços as redes tornaram-se mais complexas, exigindo requisitos de funcionamento e disponibilidade mais rigorosos. A convergência para uma infra-estrutura única e compartilhada trouxe situações que precisam ser equalizadas: aplicações de missão crítica compartilham largura de banda com tráfegos convencionais, aplicações sensíveis a atrasos concorrem com aplicações que consomem grande largura de banda e com aplicações que não suportam perdas. Todas estas situações são possíveis e a rede foi, originalmente, concebida para transporte de dados com robustez, mas sem garantias. A qualidade final dos serviços, nestas situações, é insatisfatória.

Diversas tecnologias surgiram objetivando a integração dos serviços de transporte de dados com outros serviços com requisitos de tempo real e/ou características especiais não contemplada pelas atuais tecnologias. Estas tecnologias representam diferentes abordagens para o mesmo problema. A tecnologia ATM tem uma abordagem integrada ao suporte de serviços com diferentes requisitos e características numa única rede de telecomunicações, representa a solução mais completa para o problema da qualidade de serviço, mas apresenta limitações ao uso em larga escala, sendo sua complexidade

tecnológica para otimizar banda e assegurar diferentes QoS a principal limitação. Serviços Integrados (*IntServ*) [1] e Serviços Diferenciados (*DiffServ*) [2] utilizam um modelo Internet existente e implementado, criando extensões que permitem obter qualidade de serviço utilizando ao máximo a infra-estrutura de rede existente e instalada e requerendo o mínimo de alterações estruturais. A particularidade do Protocolo de Internet (IP) é que, desde sua origem, este protocolo foi desenvolvido e implementado como um protocolo de comunicação com controle de tráfego utilizando a regra de melhor esforço *(Best-effort Service ou Lack of QoS)*, que não prevê nenhum mecanismo de qualidade de serviços e, conseqüentemente, nenhuma garantia de alocação de recursos da rede. Não se previa, na época, que a Internet fosse se tornar a grande rede mundial e que a convergência dos serviços para uma estrutura integrada fosse acontecer, mas o imprevisto aconteceu e levou os fabricantes de equipamentos de rede a desenvolverem protocolos que garantissem qualidade de serviços fim-a-fim.

As aplicações, que estão sendo desenvolvidas, são cada vez mais exigentes e são servidas pela técnica de “melhor esforço” das redes TCP/IP. Esta técnica, ainda atual, não é um problema para quem pode dispor de redes de alta velocidade, mas para quem tem recursos limitados, esta técnica é ineficiente. Os fluxos são tratados de maneira igualitária, e isto ocasiona perda de desempenho dos aplicativos vitais para as instituições nos momentos de maior concorrência. Para suprir estas necessidades, estudos foram desenvolvidos buscando a melhoria de qualidade na prestação destes serviços. O tema "Qualidade de Serviço" tem sido motivo de intensa discussão, tanto no âmbito acadêmico quanto no comercial. A globalidade do alcance da Internet, aliado à sua crescente popularidade, conduziu a Internet a ser o meio de comunicação preferido pelas novas aplicações distribuídas, como vídeo remoto, conferência multimídia, realidade virtual, dentre outras. Estas aplicações são sensíveis ao tempo de entrega dos pacotes de dados nos destinatários. A informação transmitida deve ser realizada com a menor

distorção possível. Porém, para que isto seja possível, a infra-estrutura de comunicação deve oferecer meios de garantir o provimento destes serviços dentro de limites restritos, especificados pelas aplicações.

Todas estas tecnologias desenvolvidas para resolução dos problemas citados podem ser utilizadas para dar eficiência às redes com escassez de recursos. A utilização dos recursos pode ser otimizada e os serviços essenciais podem ter tratamento diferenciado, dando ao organismo um uso mais apropriado de seus recursos. É nesta linha de raciocínio que este trabalho baseia seus principais objetivos, onde, por um lado, analisa diversas tecnologias para implementação de qualidade de serviço, por outro constrói um ambiente de serviços diferenciados (*DiffServ*) com base numa plataforma QoS Cisco (Roteadores e IOS) existente, utilizando recursos disponíveis nos ambientes de redes. O foco principal deste trabalho está voltado para o aproveitamento dos recursos presentes (e não utilizados) nos roteadores Cisco que compõem a rede da Previdência Social.

A Previdência Social é uma instituição pública que tem como objetivo reconhecer e conceder direitos aos seus segurados. A renda transferida pela Previdência Social é utilizada para substituir a renda do trabalhador contribuinte, quando ele perde a capacidade de trabalho, seja pela doença, invalidez, idade avançada, morte e desemprego involuntário, ou mesmo a maternidade e a reclusão. Enfim, a Previdência Social tem por fim assegurar aos seus beneficiários meios indispensáveis de manutenção, por motivo de incapacidade, idade avançada, desemprego involuntário, encargos de família e reclusão ou morte daqueles de quem dependiam economicamente. Com toda esta incumbência, a Previdência Social é hoje um dos maiores provedores de serviços do Brasil. São serviços essenciais a milhões de pessoas em todo território brasileiro, compreendendo cerca de vinte milhões de benefícios. São aposentados, gestantes, deficientes físicos e pessoas afastadas de seus serviços por problemas de saúde, que se deslocam com dificuldade aos postos de atendimento, muitas vezes em municípios distantes de seus domicílios, em

busca de benefícios que garantam sua sobrevivência. Estes pontos de atendimento são servidos por uma infra-estrutura de comunicação composta por enlaces de baixa velocidade, serviços de manutenção remotos, todos os servidores de aplicações são remotos e componentes de rede local de boa qualidade. A descontinuidade destes serviços ou a sua ineficiência prolonga o sofrimento destas pessoas em filas de espera e até mesmo em retornos, causando mais transtornos a estas pessoas desprovidas de recursos e de saúde. Nem todos os municípios possuem postos de atendimento, sendo necessário o deslocamento para outros centros urbanos para serem atendidos, e esta situação se torna mais crítica quando os serviços prestados pela Previdência são de baixa qualidade, necessitando de vários deslocamentos para concluir um atendimento. A previdência, também, possui seus sistemas de arrecadação que fazem o recolhimento das contribuições de pessoas físicas e empresas e presta serviços de informações de arrecadação e benefícios a todos os trabalhadores ativos, aposentados e empresas. Estes motivos, tanto de natureza social quanto de viabilidade técnica, que torna este trabalho bastante útil para a realidade das redes da Previdência e para uma melhoria na qualidade de vida dos beneficiários da Previdência Social.

A necessidade de qualidade de serviço (QoS) é bastante evidente nestes ambientes, principalmente quando se busca utilizar eficientemente os canais de comunicação de redes, que muitas vezes apresentam grande concorrência de diversos aplicativos. Com esse comportamento é imperativo fazer tratamentos diferenciados dos diversos tipos de fluxos que trafegam nessas redes, priorizando o tráfego de missão crítica em relação a outros tipos de tráfego.

Existem técnicas capazes de oferecer QoS que podemos utilizar. As principais abordagens são: O modelo de serviços integrados, o *IntServ* [1], que usa o protocolo RSVP (*ReSerVation Protocol*)[23] e faz reservas de recursos para fluxos individuais ao longo da rede, o modelo de Serviços Diferenciados, o *DiffServ* [2], que fornece QoS de forma escalável colocando a complexidade

da rede nos roteadores de borda *(edge*) e simplificando o interior (*core*) da rede. Diferentemente do *IntServ*, o *DiffServ* não necessita manter informações de estado e sinalização em cada roteador [4]. Temos, ainda, os protocolos MPLS (*Multi Protocol Labeling Switching*)[3], e o Roteamento baseado em QoS (QoSR)[4], cada um possui uma área de atuação e modo diferentes para tentar solucionar o problema da QoS.

Os roteadores *DiffServ* suportam diferentes funcionalidades, são classificados como componentes de borda ou de centro, conforme sua localização no domínio *DiffServ* (DS). Um Domínio DiffServ é composto por um grupo de nós contíguos que operam um conjunto comum de serviços [67]. Os componentes de borda localizam-se nas extremidades do domínio DS, tendo como principais funções a classificação e o Condicionamento de Tráfego (TC) de acordo com regras preestabelecidas. Os componentes de centro localizam-se dentro dos domínios DS e suas funções principais são a classificação do tráfego previamente agregado e o enfileiramento.

A fundamentação deste trabalho nasceu da análise do desempenho das aplicações da Previdência Social e das características da rede que as suporta. Verificamos que a Previdência Social é uma rede heterogênea em termos de enlaces físicos, mas homogênea em termos de equipamentos e software de rede. Esta característica foi determinante para pesquisarmos mais concentradamente uma solução para o problema de desempenho, principalmente nas localidades extremas da rede. O estudo baseado nos princípios de pouco investimento em infra-estrutura, pouco investimento em software e mudança tecnológica moderada adequou-se perfeitamente às mudanças que este trabalho sugere, pois este se aproveita de equipamentos que suportam as mudanças e sistemas operacionais que portam as facilidades requeridas.

1.1 Objetivos do Trabalho

Este trabalho elabora um estudo de viabilidade de utilização de QoS com arquitetura de serviços *DiffServ*, empregando os mecanismos de QoS em um segmento de rede com todas as principais características de uma rede da Previdência Social servida por enlace de banda estreita. Este estudo abrange todas as redes da Previdência Social, podendo ser utilizado por outras instituições com características semelhantes, fazendo a otimização do uso dos recursos destas redes, melhorando tanto os serviços prestados pela entidade quanto à qualidade de vida de seus clientes. As características das redes locais, que juntas montam a rede da Previdência Social, e o alcance social dos serviços prestados por esta rede nos remete a busca de soluções que minimizem estes problemas.

O objetivo principal deste trabalho consiste em implementar *DiffServ* em um ambiente de rede similar ao ambiente da Previdência Social, fazer ensaios de diversas configurações dos mecanismos de QoS e analisar o desempenho de cada ensaio demonstrando o ganho de desempenho com o uso de QoS no ambiente de produção.

Este trabalho mostra, através de medições, que a combinação dos diversos mecanismos em acordo com as características dos fluxos e das priorizações estabelecidas, dão um ganho significante sem onerar drasticamente o orçamento da entidade. O trabalho mostra que estes mecanismos, quando bem utilizados, racionaliza o uso da banda nos momentos de pico, dando maior eficiência aos serviços vitais da empresa.

Através de vários testes com diferentes configurações, este trabalho analisa a forma como os mecanismos de QoS podem ser configurados e utilizados nesta plataforma para depois serem transplantados para o ambiente real em conformidade com as características de cada rede.

1.2 Organização do Trabalho

Este trabalho é dividido em oito capítulos, apresenta diversas maneiras de utilização dos mecanismos de Qualidade de Serviço (QoS) em redes IP, demonstra que o uso de mecanismos de QoS traz ganhos de desempenho. Dá ênfase a redes conectadas por enlaces de baixa velocidade, onde é demonstrado que através da utilização de QoS obtem-se ganho em termos de desempenho em aplicativos prioritários e racionalidade no uso de recursos de banda.

No segundo capítulo são tratados os fundamentos das redes IP, abrangendo protocolo IP e a Internet, comutação de pacotes e as demais características e desafios das redes IP. São apresentados problemas e desafios destas redes, a tendência de utilização e os requisitos necessários para as novas aplicações.

O terceiro capítulo descreve Qualidade de Serviços, seus fundamentos, requisitos e alternativas técnicas para sua provisão através de redes heterogêneas. Este capítulo reúne todas as definições e conceitos referentes a QoS, descrevendo os requisitos essenciais, as alternativas técnicas para seu provimento, dando ênfase às soluções *IntServ* e *DiffServ*.

O quarto capítulo enfoca os mecanismos para provisão de QoS. Descreve cada mecanismo, suas funções e aplicações. É neste capítulo onde é detalhada a operacionalização da QoS, onde são descritos os algoritmos de enfileiramento, as políticas de QoS, os algoritmos gerenciamento e de contenção de congestionamento. Há descrições da moldagem e policiamento do tráfego com seus diversos métodos e controles.

O quinto capítulo trata das tarefas essenciais para provisão de serviços em um domínio *DiffServ*. Mostra o tratamento dado aos fluxos em cada etapa do provimento de QoS dentro do DS. Também descreve as funcionalidades do *DiffServ*, explica cada componente do DS, suas funcionalidades e localização, dando ênfase às particularidades da implementação Cisco.

O sexto capítulo descreve o ambiente de testes, sua montagem e composição. Descreve o testbed, mostra a metodologia aplicada para medições e a configuração dos roteadores. Compõe os testes UDP e TCP parametrizando-os de acordo com cada política escolhida. É neste capítulo que há a escolha das políticas para provimento de QoS a serem testadas e analisadas. Baseados nestas políticas são configurados os roteadores e sintonizados os mecanismos de geração e medições.

No sétimo capítulo são descritos os objetivos de cada teste, mostrado graficamente cada um, computados e analisados os resultados.

O oitavo capítulo relata as conclusões do trabalho.

2 COMUTAÇÃO DE PACOTES E REDES IP

As tecnologias de comutação (Níveis 2 e 3), denominadas genericamente de "comutação de pacotes" (*Packet Switching*), devem prevalecer como opção tecnológica para as redes de computadores e como suporte às aplicações como um todo [5]. As tecnologias de comutação mais comuns disponíveis para utilização em redes, e que não possuem maiores restrições de porte, desempenho ou cobertura geográfica (LAN - *Local Area Network*s, MAN - *Metropolitan Area Networks* ou WAN – *Wide Area Networks*) são as seguintes [6]: ATM - *Asynchronous Transfer Mode* (Nível 2), *Frame-Relay* (Nível 2) e IP (Nível 3). Considerando estas alternativas tecnológicas, existem duas formas como estas tecnologias podem ser utilizadas pelas aplicações. Na primeira forma a aplicação utiliza diretamente a comutação de pacotes, como ATM e *Frame-Relay,* e pode eventualmente utilizar recursos do IP ou utilizá-lo diretamente. Um exemplo desta modalidade é a utilização da tecnologia ATM como *backbone* de rede. Na segunda forma a comutação no IP é a plataforma para as aplicações. Nesta situação, o IP é responsável pela comunicação, podendo ser utilizado sobre algumas tecnologias de nível 2 como ATM e *Frame-Relay*, sendo estas tecnologias, nesta situação, meros mecanismos de transporte de pacotes entre roteadores.

2.1 O Protocolo IP e a Internet

O protocolo IP foi desenvolvido e implementado como um protocolo de comunicação com controle de tráfego utilizando a regra do melhor esforço (*Best-effort Service ou Lack of QoS*), que não provê nenhum mecanismo de qualidade de serviços e, conseqüentemente, nenhuma garantia de alocação de recursos da rede [7]. A projeção na época, não vislumbrava que a Internet se tornaria a grande rede mundial que hoje é conhecida. E, desse rápido crescimento da Internet, a tendência atual é a integração de todos os dados numa única infra-estrutura de redes de pacotes, a rede IP.

As redes IP têm uma grande base instalada com milhares de computadores que continua crescendo em todos os continentes. O crescimento da Internet e a grande aceitação da base tecnológica TCP/IP pelas empresas são os fatores que continuam a impulsionar o crescimento e a aceitabilidade das redes IP. Redes como de telefonia, de rádio e de televisão também estão convergindo para IP aproveitando-se de sua abrangência e flexibilidade. Este quadro não tende a mudar nos próximos anos e, assim sendo, teremos cada vez mais computadores utilizando o TCP/IP para suas comunicações na Internet e no âmbito das empresas.

2.2 Roteamento IP

A simplicidade é um dos motivos para o sucesso do protocolo IP. O projeto do IP é baseado em um princípio bem simples derivado do argumento "*end-to-end*" que coloca a inteligência nos nós extremos da rede (nó fonte e nó destino) deixando os nós do núcleo da rede com pouca inteligência. Isto porque os roteadores ao longo da rede apenas conferem o endereço destino do pacote IP, comparam com uma tabela de roteamento para determinar o próximo nó (*next hop*). Se a fila estiver longa para o próximo nó, o pacote por ser atrasado. Se cheia ou indisponível, o roteador pode descartar o pacote.

O IP provê um serviço do tipo "melhor esforço" e pode acontecer perda de dados e/ou atrasos imprevistos.

A tarefa de roteamento IP é executada pelo protocolo IP. Este protocolo é responsável pela entrega das informações geradas pelas aplicações aos seus destinos de forma correta e eficiente. Esta tarefa tem sua execução baseada em tabelas de roteamento existentes nos equipamentos conectados à Internet, onde constam as rotas a serem seguidas pelas informações em função do seu destino. A manutenção dessas tabelas é descentralizada na Internet, via combinação das seguintes formas:

- Forma estática - O administrador de uma dada rede pode configurar manualmente rotas para todas as redes a ela ligadas, e uma rota

‘*default*’ (*default gateway*) ‘apontando’ para o próximo roteador em direção ao núcleo da Internet.

- Forma dinâmica – Nesta forma o administrador pode fazer uso de algumas aplicações que implementem protocolos de roteamento, que automaticamente determinam e divulgam essas rotas. A grande vantagem do uso de protocolos de roteamento sobre a forma anterior é a possibilidade de adaptação dinâmica das rotas a situações de falhas ou congestionamentos detectados.

A Internet é composta por Sistemas Autônomos[1], sendo que dentro desses sistemas as rotas são definidas de forma estática ou através de protocolos interiores de roteamento, e um de seus equipamentos é eleito para divulgar essas rotas para outros sistemas autônomos via protocolos exteriores de roteamento. Dentre os protocolos de roteamento mais usados destacamos, como protocolos interiores: o RIP - *Routing Information Protocol*, que utiliza distância vetorial para comparar matematicamente rotas, indicando o melhor caminho para um endereço de qualquer destino, e o OSPF - *Open Shortest Path First*, que é um protocolo de roteamento do tipo *link-state*, que envia avisos sobre o estado da conexão (*link-state advertisements*, LSA) a todos os outros roteadores em uma mesma área hierárquica e usa o algoritmo SPF para calcular a menor rota para cada nó; como protocolos exteriores temos, o EGP - *Exterior Gateway Protocol*, que é um protocolo que informa a um dispositivo de rede IP como alcançar outras redes IP, dando-lhe a direção que a rede desejada está; e o BGP-4 *Border Gateway Protocol version 4*, que serve para informar às redes externas a um sistema autônomo, quais são as rotas para redes atingíveis dentro de sua rede.

[1] *Sistema autônomo* geralmente é uma rede e suas subredes associadas ou uma coleção de redes e subredes sob uma mesma administração.

2.3 Aplicação sobre Redes IP

Nascida e desenvolvida tendo como um dos principais objetivos o requisito de poder ser utilizada com os diversos tipos de meios físicos e tecnologias existentes na época de sua criação ("IP sobre Tudo" - Anos 70) de forma a viabilizar a comunicação entre as aplicações fim-a-fim em rede, a rede TCP/IP é capaz de comutar sobre meios físicos e tecnologias de nível 2 confiáveis, não-confiáveis, de alto desempenho e de baixo desempenho. A arquitetura do protocolo IP foi concebida de maneira simples visando atender o cenário imaginado na época para sua implantação em termos de rede. Devido a esta simplicidade, o paradigma de concepção impõe algumas restrições técnicas ao IP e, por conseqüência, submete todas aplicações às mesmas limitações das aplicações com poucos requisitos de operação. Nesta sentido, aplicações de dados que podem perder pacotes ou permitir a existência de atrasos.

Hoje, devido às exigências de utilização das redes, é necessário que qualquer aplicação possa executar com qualidade sobre o IP. Ou seja, a situação do IP atualmente é no sentido do "Tudo sobre IP" mantendo a premissa básica de projeto do "IP sobre Tudo" dos anos 70.

2.4 Convergência para IP

A convergência para IP nas redes de dados já é uma realidade. Além das redes de dados, já aconteceram migrações de muitas tecnologias de informação e aplicações comerciais e outras estão em mudança. A realidade é que em termos de exigência de rede, a maioria das aplicações possui baixa prioridade e baixo requisito de largura de banda e alta tolerância a atrasos, mas outras têm rígidas exigências operacionais. Por exemplo, a migração das redes de rádio e televisão ainda necessita de uma evolução tecnológica. Primeiramente, é necessário aumentar a largura de banda, e isso já está acontecendo. O modelo de difusão em *broadcast* (de um para todos) atual deve evoluir para o modelo IP *multicast*, (de um para muitos), onde há um

conhecimento do destino, satisfazendo as características destas redes. O modelo *unicast* (de um para um) na Internet nunca poderia atingir os níveis abrangência que os serviços de difusão *broadcast* atingem. Por este motivo, é necessário o desenvolvimento de *multicast* para habilitar a convergência de redes de televisão e rádio para IP.

As redes IP podem prover para aplicações de áudio e vídeo novas habilidades, dando novas dimensões ao conteúdo da multimídia, tais como[8]: inclusão de ligações WEB ou, simultaneamente, envio de *slides* e arquivos ou outro conteúdo durante a transmissão para enriquecer a entrega, e comunicação nos dois sentidos, permitindo aos receptores de conteúdo interagir com provedores e, como *multicast* permite aos receptores falarem entre si na forma de muitos-para-muitos.

As redes IP abrem a porta a uma nova geração de aplicações com novas possibilidades, tais como: grupos de discussão, ensino à distância com interatividade, jogos e espetáculos, e pesquisas instantâneas.

Inegavelmente as redes IP possuem um grande potencial para audiências globais, de forma econômica, e de acordo com as características específicas de cada seguimento. Mas há outras exigências de serviços de rede para satisfazer além de *multicast*. Requisitos de tempo para aplicações interativas em tempo real, como telefonia sobre IP, são mais significativos que os de largura de banda. A falta de qualidade na conversação entre dois interlocutores tem evidências imediatas sobre a qualidade ou falta de qualidade em uma chamada. Perdas e atrasos são percebidos e comprometem a qualidade do diálogo. Atrasos superiores a 500ms na entrega fim-a-fim, podem inviabilizar a chamada. Devido a sua imprevisibilidade, o efeito do tráfego em rajada em uma rede congestionada acasiona atrasos no roteamento e perdas de pacotes resultando em tempos de ida e volta inadequados em redes IP, o que limita severamente a utilização de serviços de telefonia sobre IP.

O incremento na largura de banda melhora o serviço IP resolvendo o problema do atraso, mas, por razões de custos ou disponibilidade destes

serviços pelas concessionárias de telecomunicações, nem sempre isto é possível. Entretanto, somente este incremento não é suficiente para satisfazer os requisitos de aplicações como a de telefonia.

2.5 Desafios - Redes IP

Concebido como um protocolo simples, o IP não tem praticamente nenhuma garantia de qualidade de serviço. A base instalada do IP é muito grande, o que torna a mudança do protocolo uma idéia pouco factível. Estas características conduzem estas redes IP a dois desafios importantes. Primeiramente o IP tem um desafio de caráter técnico. Por exemplo, o IP não garante vazão constante para uma aplicação em particular. Além disso, uma aplicação não tem garantia de entrega de pacotes que eventualmente são descartados ou perdidos sem que nenhum tipo de correção ou ação seja tomada. Não existe nenhuma garantia de tempo de entrega para os pacotes. O outro desafio está voltado para a maneira de como adaptar o protocolo e manter as mesmas características mesmo com a mudança do paradigma. Embora com a previsão de mudança do IP (Versão 4) para o IP (Versão 6) [9], permanece o paradigma de simplicidade inicial do IPv4 para o IPv6. O Ipv6 (I*png - IPNew Generation*) trata de outras questões de implementação do protocolo (Endereçamento, segurança, autoconfiguração) e não traz nenhuma solução completa para os desafios apresentados.

Para suprir as limitações do protocolo IP é necessário que sejam propostos novos protocolos, algoritmos e mecanismos que tratem estas deficiências e permitam o suporte efetivo de qualquer tipo de aplicação sobre redes IP.

2.6 Novas Aplicações

Um universo de aplicações está surgindo e o IP, com uma base instalada muito grande, tende a suportar estas novas aplicações em rede. São, em sua

maioria, aplicações multimídia, pois elas envolvem a transferência de múltiplos tipos de mídia (dados, vídeo, voz, gráficos,...) com necessidades de tempo e sincronização para a sua operação com qualidade. Dentre estas aplicações destacamos[7]:

- Telefonia e Fax sobre IP (VoIP - *Voice over IP*)
- Comércio Eletrônico (*E_commerce)*
- Vídeo sobre IP
- Educação à Distância (EAD) *(Distance Learning)*
- Vídeo-Conferência
- Aplicações Colaborativas e de Grupo
- Aplicações Multimídia
- Aplicações Tempo Real

2.7 Problemas das redes IP

O protocolo IP apresenta algumas limitações, fruto de suas simplicidade original, que limitam a implementação de QoS nas redes baseadas neste protocolo. Entre estas limitações podemos citar [10]:

- O IP é um protocolo sem conexão, não havendo mecanismos de controle de admissão;
- A rede só fornece serviço do tipo "melhor esforço", sem haver nenhum tipo de classificação, priorização e reserva de recursos;
- É adotado o esquema de roteamento implícito, onde cada pacote é inspecionado e analisado individualmente em cada nó de roteamento, acarretando uma sobrecarga na transmissão dos mesmos.

Por causa do crescimento acentuado das redes IP, um dos principais

problemas enfrentado por estas redes é a questão do compartilhamento dos recursos [6]. Como os pacotes entram na rede de maneira assíncrona, mais de um pacote pode entrar em um nó exatamente ao mesmo tempo e serem destinados à mesma saída. Este fato ocasiona a disputa pela capacidade de processamento do nó, pois estes pacotes não serão tratados simultaneamente e alguns terão que esperar até os outros sejam processados. Haverá formação de filas e pacotes terão que ser armazenados em *buffer*, ou até mesmo descartados se o tráfego exceder a capacidade de rede e processamento do nó por um tempo excessivo.

2.8 O protocolo UDP

O *User Datagram Protocol* é um protocolo de transporte que faz uso dos mecanismos definido pelo RFC768 [70]. Toda mensagem UDP enviada à rede trafega na área de dados de um datagrama IP, como mostrado a seguir. O mecanismo de comunicação do UDP é bastante simples , este mecanismo provê um serviço sem conexão, não confiável, sem utilização de confirmação e ordenação de mensagens e não oferece um mecanismo de controle na velocidade de envio de mensagens. Por este motivo, as mensagens UDP podem sofrer problemas na entrega, duplicação, retardo, e podem chegar tão rapidamente que o destino não consegue processá-las. Para muitas situações este protocolos é a melhor solução e muitos *softwares* o utilizam. A figura 2.1 mostra o encapsulamento de um pacote UDP na área de dados de um datagrama IP.

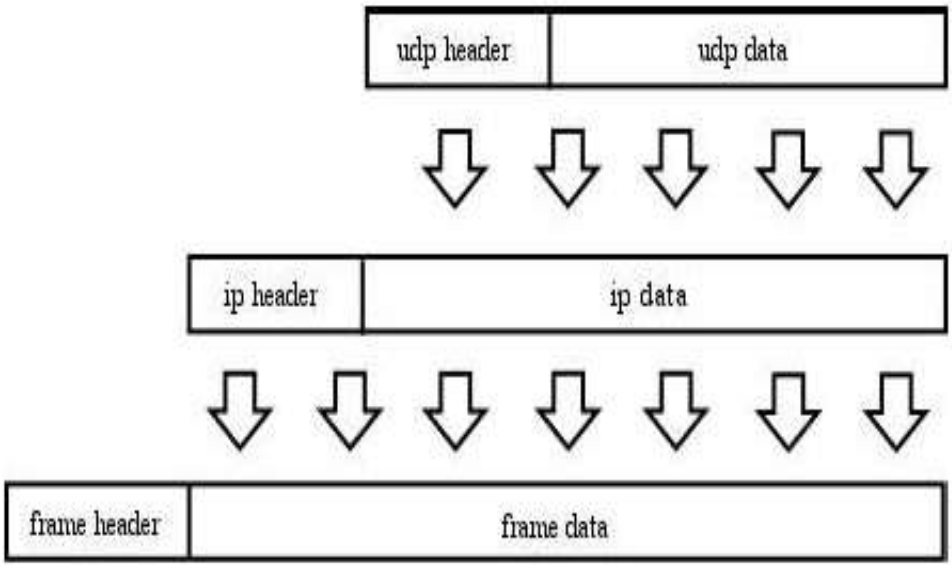


Figura 2-1 Encapsulamento UDP na área de dados de um datagrama IP.

A principal característica do protocolo UDP, presentes também em outros protocolos de transporte, está no fato de vários processos de usuário poderem estar acessando simultaneamente serviços de rede sem sofrer qualquer tipo de *interferência.*

A confiabilidade da implementação do UDP é totalmente direcionada às aplicações. É comum encontrar *software* que utilizam UDP e possuem mecanismo próprio para controle de perdas de mensagem, duplicação, etc. O baixo custo para a rede é o principal motivo da utilização do UDP pelas aplicações. Devido ao formato extremamente simples da mensagem deste protocolo, o processamento de máquina para seu envio ou entrega é bastante reduzido.

2.9 O protocolo TCP

O *Transmission Control Protocol* é um protocolo de transporte que faz uso dos mecanismos definido pelo RFC793 [71]. É o responsável pelo serviço de transporte de fluxo confiável na família TCP/IP. O mecanismo TCP provê um serviço orientado à conexão, confiável, com utilização de confirmação e ordenação de mensagens e oferece um mecanismo de controle na velocidade

de envio de mensagens. O TCP especifica todos os detalhes necessários para oferecer uma transferência confiável, que vão desde o estabelecimento de uma conexão até a confirmação de um simples pacote recebido durante uma transferência. Assim como o UDP, o TCP é um protocolo de transporte o que significa que todas suas mensagens são encapsuladas na área de dados de um datagrama IP antes de serem enviadas à rede, conforme mostrado pela figura 2.2.

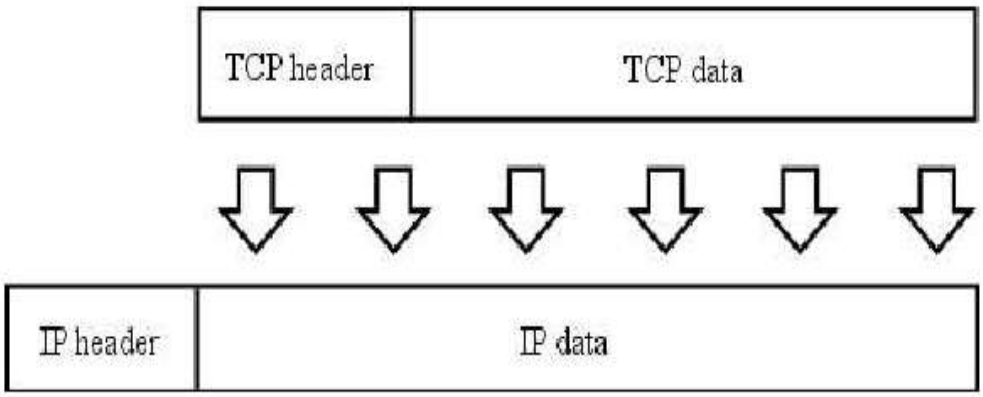


Figura 2-2 - Encapsulamento TCP na área de dados de um datagrama.

Umas das funções dos protocolos de transporte é proporcionar um método de comunicação entre aplicações em duas extremidades. O TCP utiliza o conceito de *portas de protocolo*[1], mas como ele é um protocolo orientado à conexão, ele usa um par de *pontos terminais*[2] para identificar uma conexão.

O conceito de janela deslizante é muito importante na transmissão de fluxo. Esta técnica é responsável pela eficácia na transmissão de *stream*. *A* figura 2.3 mostra graficamente este conceito.

[1]Local associado ao *software* onde a rede envia ou busca datagramas de forma independente.
[2]Conjunto de endereço IP do *host* e porta utilizada na conexão.

Figura 2-3 - Janela deslizante

O mecanismo inicia o funcionamento quando um transmissor envia um pacote, então é iniciado um temporizador que espera uma confirmação. Após receber esta confirmação ele desliza na janela e transmite outro pacote. Os dados são transmitidos apenas em uma direção, isto pelo fato de que quando um pacote é enviado, a rede fica inativa a espera de uma resposta. Quando o pacote chega no receptor, o mesmo envia um pacote de confirmação, e volta a inatividade esperando outro pacote para que possa confirmar. Por este modelo, a transmissão seria lenta e a utilização da largura de banda extremamente baixa. Este é o princípio de funcionamento dos protocolos que implementam janelas deslizantes.

Para melhorar o uso da largura da banda, estes protocolos podem enviar diversos pacotes ao mesmo tempo antes de esperar uma confirmação. A medida que as confirmações vão chegando novos pacotes vão sendo enviados, ou seja, a janela vai deslizando. A figura 2.4 ilustra melhor o conceito do envio de vários pacotes sem confirmação. A eficiência e desempenho dos protocolos de *janela deslizante* ficam dependentes do tamanho da janela e conseqüentemente da velocidade de processamento da rede. Uma rede de alta velocidade ao contrário de uma rede de baixa velocidade, por exemplo, suporta um tamanho de janela alto, fazendo com que as transmissões sejam feitas mais rapidamente.

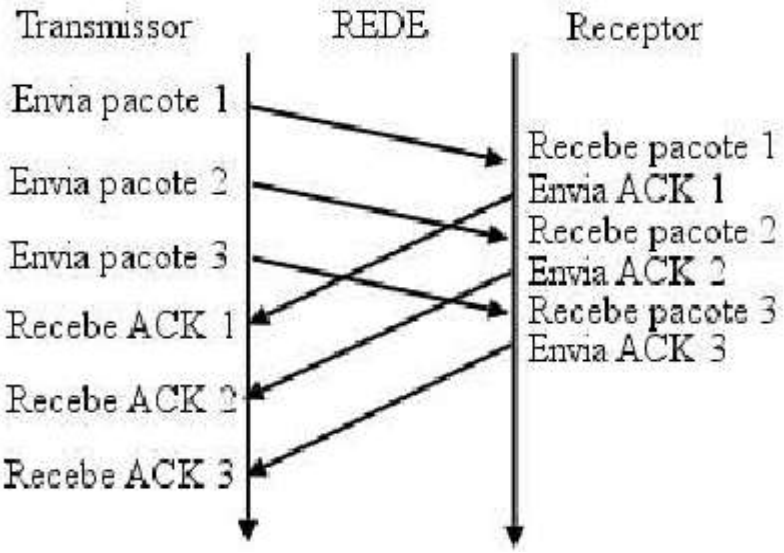


Figura 2-4 - Transmissão simultânea pacotes sem aguardar confirmação.

O mesmo modelo de janela é mantido da extremidade receptora, para que se tenha controle das confirmações sob pacotes que foram recebidos. Nós podemos interpretar a janela, conforme a figura 2.3, de forma que os pacotes à esquerda da janela significam que foram confirmados, os que estão dentro da janela estão sendo transmitidos e os da direita, ainda não foram transmitidos.

No TCP o conceito de janela flutuante é um pouco diferente e oferece dois serviços importantes [73]: transmissão eficaz e controle de fluxo. A questão relacionada a transmissão eficaz está diretamente relacionada a utilização da largura de banda. Assim como o modelo que vimos anteriormente, o TCP envia vários segmentos para rede antes que uma confirmação chegue. O controle de fluxo é outro serviço oferecido pelo TCP onde o receptor consegue definir um tamanho de janela de acordo com o espaço livre em seu *buffer*.

Uma das características mais importantes que o TCP suporta é a característica de janelas de tamanho variável. Existe um campo especial no cabeçalho TCP que permite ao receptor determinar quantos octetos espera receber. Este valor pode aumentar, diminuir ou permanecer o mesmo. No

primeiro caso, o transmissor aumenta o tamanho de sua janela e continua a enviar octetos ao receptor. Se a extremidade transmissora receber um notificador de diminuição de janela, o mesmo processo é realizado, entretanto, a redução não pode chegar ao limite daqueles octetos que já foram previamente definidos para envio em uma outra ocasião. Com um tamanho de janela variável o TCP consegue controlar o fluxo de pacotes, seja ele para um aumento ou redução no nível de emissão de octetos. Uma conseqüência muito importante deste mecanismo é conseguir geralmente um uso considerável da largura de banda. Através disso também, uma máquina consegue enviar ao emissor um notificador de janela menor quando verificar que seu *buffer* está ficando cheio.

O TCP utiliza como unidade de transferência básica o segmento , assim como o IP utiliza o datagrama. Um segmento TCP além de carregar dados, ele pode ser utilizado para estabelecer e encerrar conexões, informar tamanho de janela, enviar confirmações, entre outras coisas.

Os segmentos transferidos entre dois pontos muitas vezes tem tamanho diferentes. Por este motivo, duas extremidades precisam combinar o tamanho de segmento que vão utilizar, pois pode ocorrer que máquinas que tem pouco espaço de *buffer* receberem um número muito grande de segmentos quando conectadas a grandes computadores. Para a negociação do Tamanho Máximo de Segmento (*Maximum Segment Size MSS)* entre origem e destino, o TCP usa o campo *opções*. O RFC879 [72] fala sobre tamanhos máximos de segmentos. É comum as máquinas em um ambiente de rede local, utilizarem como base de MSS a MTU da rede ou podem utilizar alguma forma de mapeamento para ver qual a menor MTU existente da origem até o destino, e definem o MSS em cima deste valor. Não é uma tarefa trivial escolher um tamanho máximo de segmento que irá trafegar entre redes interligadas. Há duas escolhas mais comuns neste propósito: tamanhos pequenos e tamanhos grandes de segmento. Um pequeno tamanho pode ser bom para fragmentação, pois os pacotes provavelmente não serão fragmentados, mas há um comprometimento na da

largura de banda. Uma outra opção é escolher pacotes consideravelmente grandes. Nesta situação é necessário lembrar que todo segmento TCP é encapsulado na área de dados de um datagrama IP que é encapsulado num *frame* de rede. O IP precisa fragmentar pacotes quando os datagramas grandes trafegam em redes com MTU baixa. Um fragmento de um datagrama não tem *vida própria* e não pode ser confirmado ou retransmitido isoladamente. Se um fragmento se perde, todo datagrama IP tem que ser retransmitido. Esta possibilidade deve ser considerada no ambiente de interligação em redes. Nesta situação, aumentar o tamanho de um segmento pode não ser solução. O ideal para o tamanho máximo de segmento seria aquele de maior tamanho possível estando livre de fragmentações durante todo o percurso. Esta situação é muito difícil na prática, principalmente pelo fator *roteamento dinâmico*. O tamanho definido em um instante para uma transmissão pode ser totalmente diferente de um outro tamanho para o mesmo destino em outro instante. O mais comum é que redes que estão diretamente conectadas utilizam um MSS próximo a MTU da rede. As redes não locais geralmente optam pelo tamanho padrão de 536 octetos. Entretanto, atualmente este valor é na maioria das vezes próximo de 1500. É necessário maiores cuidados ao classificar redes locais e não- locais. Redes com mesmo *network ID* e diferenciando-se apenas na porção de *subnet ID* entram para o conjunto de redes locais, enquanto as redes bem distintas quanto ao *network ID* são consideradas redes não locais.

Uma característica muito importante e que necessita de atenção no TCP é o fato de o mesmo ter mecanismos de reação à congestionamentos. Com isso o TCP consegue prever e se recuperar (dentro do possível) de um congestionamento.

2.10 Considerações sobre este capítulo

Este capítulo tratou da tecnologia de comutação de pacotes e sua projeção de uso futuro. Tratou, particularmente, das redes IP, começando pelas características de concepção do protocolo IP e sua atual utilização. Abordou toda herança criada por esta tecnologia e as necessidades que deverão ser

supridas para seu uso futuro. Explicou algumas formas de roteamento. Mostrou a crescente convergência para redes IP, além das novas aplicações utilizando esta tecnologia. Depois tratou dos desafios que as redes IP terão que enfrentar, as necessidades das novas aplicações e dos problemas existente nesta tecnologia.

O capítulo seguinte apresenta uma visão abrangente de Qualidade de Serviços englobando seus avanços e requisitos necessários à sua implementação.

3 QUALIDADE DE SERVIÇO – DEFINIÇÕES E CONCEITOS

Quando a Internet trabalhava apenas com aplicações simples, a demanda era bem atendida pelo nível de qualidade oferecido. Com o crescimento progressivo da Internet nestes últimos anos e o seu conseqüente amadurecimento, surgiram novas aplicações distribuídas, que necessitavam de grande largura de banda e eram sensíveis a problemas de atrasos. Desta maneira, diversas classes de serviços com necessidades de recursos e diferentes prioridades passaram a ser demandadas gerando a necessidade de melhoramentos nos serviços oferecidos. Desta necessidade de melhoria dos serviços oferecidos surgiu a idéia do paradigma conhecido como Qualidade de Serviço (QoS).

3.1 Que é Qualidade de Serviço (QoS)

Qualidade de Serviço não é resumida a uma única expressão. QoS tem conotações diferentes para diversas áreas. De início, QoS pode ser entendido como sendo um requisito das aplicações, que exige que determinados parâmetros (atrasos, vazão, perdas, etc) estejam dentro de limites bem definidos (valor mínimo e valor máximo). No entanto, a garantia de QoS, em redes de computadores, envolve diversos níveis de atuação em vários tipos de equipamentos e tecnologias, assim, esses parâmetros não estão localizados em apenas um único equipamento ou componente da rede. Considerando esse fato, a Qualidade de Serviço deve atuar em todos equipamentos, camadas de protocolo e entidades envolvidas [66].

A realidade é que Qualidade de serviço (QoS) assume significados diferentes para pessoas diferentes, tornando-se um dos tópicos mais confusos e difíceis de definir em redes de computadores hoje em dia [10]. Algumas definições são:

ISO: Na visão da ISO, QoS é definida como o efeito coletivo do desempenho de um serviço, o qual determina o grau de satisfação de um

usuário do serviço [12].

Sistemas Multimídia Distribuídos: Em um sistema multimídia distribuído a qualidade de serviço pode ser definida como a representação do conjunto de características qualitativas e quantitativas de um sistema multimídia distribuído, necessário para alcançar a funcionalidade de uma aplicação [13].

Redes de Computadores: QoS é utilizado para definir o desempenho de uma rede relativa às necessidades das aplicações, como também o conjunto de tecnologias que possibilita às redes oferecer garantias de desempenho [14]. Em um ambiente compartilhado de rede, QoS necessariamente está relacionada à reserva de recursos ou priorização de tráfego. Essa reserva pode ser feita para um conjunto (agregação) de fluxos ou sob demanda para fluxos individuais. QoS pode ser interpretada como um método para oferecer alguma forma de tratamento preferencial para determinada quantidade de tráfego da rede. Em outra definição, QoS é vista como a capacidade de um elemento de rede ter algum nível de garantia de que seus requisitos de serviço e tráfego podem ser satisfeitos [15].

Na ótica dos usuários [5], Qualidade de Serviço para uma aplicação pode ser variável, e a qualquer momento, pode ser modificada (para melhor ou pior qualidade). O ajuste de QoS adequada é um requisito de operação da rede e de seus componentes para viabilizar a operação com qualidade. Assim, QoS é garantida pela rede através de ajustes de seus componentes e equipamentos.

Na visão dos programas de aplicação, QoS é expressa e solicitada em termos de "Solicitação de Serviço" ou "Contrato de Serviço" [5]. A solicitação de QoS da aplicação é denominada de SLA - Contrato de Nível de Serviço (*Service Level Agreement*), que tem como objetivo especificar os níveis mínimos de desempenho que um provedor de serviços deverá manter a disposição do usuário e o descumprimento desse acordo implica em penalidades estipuladas em contrato.

Conforme [16], Qualidade de Serviço refere-se à habilidade da rede em prover um melhor serviço para um determinado tráfego em relação aos outros tráfegos concorrentes. Isto é conseguido através de garantia de banda, na diminuição de perdas de pacotes, no gerenciamento e prevenção de congestionamento da rede, na limitação de banda para determinadas classes de tráfego e na priorização de tráfego através da rede. Este é o conceito mais claro e objetivo, que mostra os parâmetros a serem trabalhados na obtenção QoS. Com isso é possível visualizar os mecanismos de provisão, presentes na rede da Previdência, que serão manipulados para atingir cada necessidade de QoS.

Neste trabalho, que enfoca as necessidades da Previdência, QoS está bem relacionada com o uso otimizado dos recursos rede, envolvendo uma classificação bem criteriosa para priorização de determinados fluxos, a racionalização do uso da banda e o uso de restrições que garantam que os serviços prioritários estejam dentro dos limites aceitáveis de operação, mesmo em detrimento de outros serviços de menor relevância.

Assim, Qualidade de Serviço é uma conjunção de equipamentos e tecnologias que possibilitam às redes garantirem desempenho às aplicações.

3.2 Provimento de QoS através de rede heterogênea

O provimento de QoS através de rede heterogênea é conseguido através de três componentes [16]: QoS em um simples elemento de rede, que inclui enfileiramento (*queueing*), escalonamento (*scheduling)* e formatação de tráfego (*traffic Shaping);* técnicas de QoS para identificação e marcação de pacotes para um serviço fim-a-fim entre elementos de rede e; policiamento e gerenciamento de QoS para controlar e administrar tráfego fim-a-fim através da rede.

3.2.1 QoS em Simples Elemento de Rede

Gerenciamento de congestionamento, gerenciamento de fila, eficiência de *link*, e policiamento e formatação de trafego (*policing and shaping)* são ferramentas para implementação de QoS em um simples elemento de rede. O gerenciamento de congestionamento é necessário para que seja evitado o congestionamento ocasionado pelos fluxos das aplicações. Este mecanismo escolhe a maneira de tratar o fluxo, colocando os pacotes em diferentes filas de prioridades ou colocando em uma única fila, onde todos os fluxos recebem o mesmo tratamento. O gerenciamento de fila é necessário para racionalizar o uso deste recurso e não permitir que filas encham e provoquem transbordo (*overflow)*, o que é bastante indesejável, e os pacotes excedentes sejam descartados (*tail drop*) ou tenham um tratamento de descarte com regras de prioridades estabelecidas. O algoritmo *Weighted Random Early Detected* (WRED), que será descrito posteriormente, pode prover mecanismos para tratar estas duas situações.

3.2.2 Identificação e Marcação

A identificação e marcação são conseguidas através do processo de classificação. Este processo consiste em provê um serviço preferencial a um determinado tipo de tráfego, que deve ser identificado principalmente por um ou vários campos do cabeçalho IP, pela interface de entrada ou por *Access Lists*, e depois pode ou não ser marcado. Se o pacote é identificado e não é marcado, é dito que pertence a uma *per-hop basis*, pois esta classificação pertence apenas a este nó, não passando para outro nó. Quando o pacote é classificado e marcado, o bit *ip precedence* é atualizaado e a marcação acompanha o pacote até o destino ou até uma nova marcação.

3.2.3 Formatação de Tráfego e Policiamento(*Traffic Shaping e Policing)*

A formatação do tráfego (*shaping)* é usada para limitar o tráfego para dentro de uma determinada largura banda. Isto é usado para prevenir transbordo de filas(*overflow)* em muitos casos. Um exemplo é quando a fonte

tem um *link* com banda mais larga do que o destino. Neste caso, podemos colocar formatação(*shaping)* para que o fluxo esteja dentro da largura de banda do destino e evite a situação de transbordo(*overflow)*. O policiamento (*policing)* oferece um serviço similar à formatação(*shaping)* diferenciando na maneira de tratar o excesso, que normalmente é descartado.

A comunidade científica se empenhou em pesquisas de desenvolvimento de novos mecanismos de suporte a aplicações avançadas para atender a demanda por QoS na Internet. Alguns mecanismos já foram propostos e padronizados e estão sendo utilizados pelos diversos fornecedores de soluções de QoS. As iniciativas do IETF (*Internet Engineering Task Force*) foram as que apresentaram um melhor conjunto de soluções para mecanismos de controle de QoS. Estas soluções são consideradas basicamente em quatro abordagens[17]: classificação do tráfego por priorização (*DiffServ*); reserva prévia de recursos (*IntServ*); melhoramento do encaminhamento (MPLS) e melhoramento das técnicas de roteamento (QoSR). Estas alternativas estão descritas nos próximos itens.

3.3 Os Avanços da QoS em Redes IP

Baseado em datagramas, o protocolo IP oferece um serviço sem conexão, que não garante a entrega dos mesmos a tempo, não garante que eles cheguem ao destino na ordem correta e nem mesmo garante que cheguem ao destino. Os roteadores, mesmo utilizando algoritmos otimizados, não podem oferecer garantias a respeito da entrega dos pacotes. Esse tipo de serviço sem conexão é conhecido como serviço de melhor esforço (*Best Effort*). No serviço de melhor esforço, a rede tenta encaminhar todos os pacotes o mais rápido possível, mas não pode fazer qualquer tipo de garantia quantitativa sobre a Qualidade de Serviço (QoS)[18].

Geralmente, QoS especifica o grau de satisfação ou visão do usuário com relação à prestação de um serviço. QoS pode ser definida quantitativamente em termos de parâmetros como, por exemplo, taxa de perda, atraso fim a fim, e *jitter* (variação do atraso). Para aplicações de tempo real, tais como:

videoconferência, educação à distância e telemedicina, QoS é fundamental para estes tipos de negócios. Em função disso, uma intensa atividade de pesquisa vem sendo conduzida no desenvolvimento de protocolos e arquiteturas para suportar tanto o tráfego tradicional, quanto o tráfego multimídia e de tempo real na Internet.

Os esforços para viabilizar QoS fim a fim para redes IP proporcionaram o desenvolvimento de várias alternativas técnicas, que trataremos a seguir.

3.3.1 Alternativas Técnicas para Provisão de QoS em Redes IP

A grande maioria das alternativas técnicas são iniciativas do *IETF (Internet Engineering Task Force*). O IETF está fortemente empenhado em propor um conjunto de soluções para os mecanismos de controle de QoS que garanta a interoperabilidade dos mesmos entre diferentes fornecedores. Isto se dá em função da importância das redes IP para o suporte de novas aplicações multimídia, tempo real, etc. Dentre as alternativas técnicas básicas desenvolvidas para oferecer qualidade de serviço em redes IP, destacamos as seguintes:

- *IntServ - Integrated Services Architecture* com o RSVP (*Resource ReSerVation Protocol*);
- *DiffServ - Differentiated Services Framework*;
- MPLS - *MultiProtocol Label Switching*;
- Roteamento baseado em QoS (QoSR) e
- Dimensionamento.

Além destas alternativas, ainda existem as soluções proprietárias.

Todas estas alternativas apresentam seus princípios, mecanismos específicos e estratégia, as quais são apresentadas a seguir.

3.3.1.1 Serviços Integrados (*IntServ*)

A arquitetura *IntServ* [1] usa um mecanismo explícito para sinalizar requisitos de QoS por fluxo para os elementos da rede *(hosts* e roteadores). Esta arquitetura é definida pelo IETF e corresponde a um conjunto de recomendações (*RFCs - Request for Comments*) visando a implantação de uma infra-estrutura robusta para a Internet que possa suportar o transporte de áudio, vídeo e dados em tempo real além do tráfego de dados transportado na infra-estrutura atual.

O princípio arquitetural desta arquitetura é que a qualidade de serviço (QoS) na arquitetura *IntServ* é garantida através de mecanismos de reserva de recursos na rede.

Neste modelo temos alocação para dois tipos de serviços, além do "Best Effort":

Serviços Garantidos - para aplicações que exigem limites fixos de atraso.

Serviços de Carga Controlada – para aplicações que exigem um limite probabilístico de atraso.

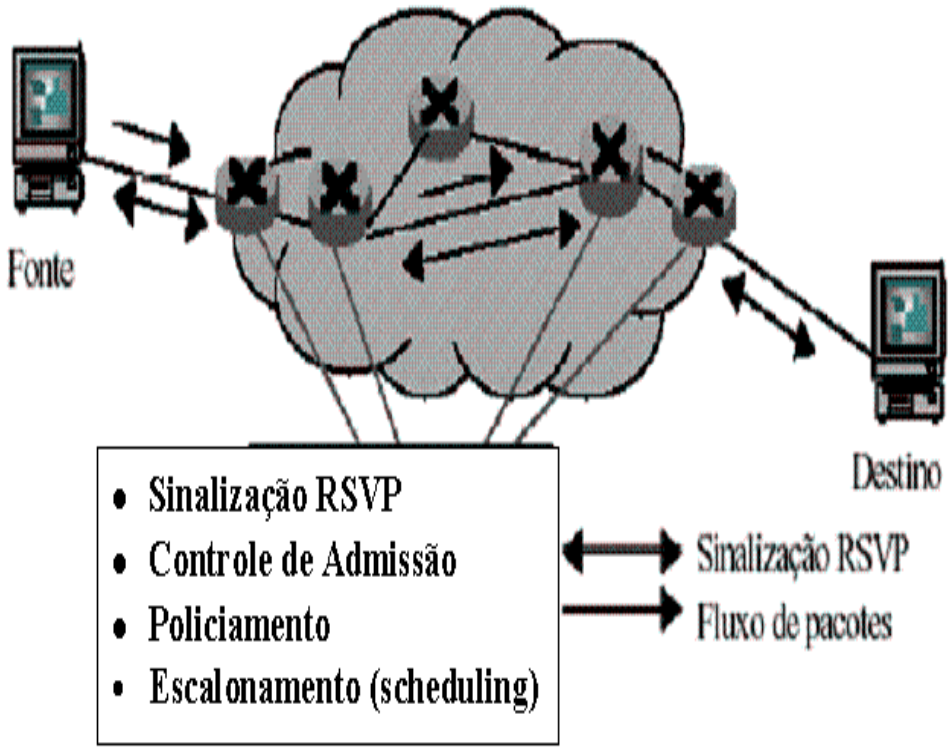


Figura 3-1 – Arquitetura de Serviços Integrados *(IntServ)*

A Figura 3-1 mostra, de maneira simplificada, a arquitetura *IntServ*, que inclui os seguintes componentes:

• Escalonador de pacotes (*scheduling*): este componente gerencia o encaminhamento dos diversos fluxos utilizando alguma disciplina de filas, por exemplo, WFQ (*Weight Fair Queueing*) [19];

• Controle de admissão de recursos: implementa o algoritmo utilizado pelo roteador para determinar se a solicitação de QoS de um novo fluxo pode ser atendida sem interferir nas garantias que outros fluxos receberam.

• Protocolo de sinalização[1] (RSVP – *Resourse ReSerVation Protocol* [20]): usado por uma aplicação para informar à rede seus requisitos de QoS e efetuar a reserva de recursos ao longo do caminho que o pacote irá percorrer.

• Policiamento: verifica se o fluxo está de acordo com as especificações negociadas na fase de estabelecimento da conexão. Fluxos fora do acordo

[1] Em princípio, o modelo *IntServ* pode usar outros protocolos de reserva, mas na prática o RSVP é o padrão de fato.

podem ter seus pacotes descartados para evitar congestionamento.

No mecanismo desenvolvido para a arquitetura *IntServ* os recursos necessários a uma aplicação são reservados antes de iniciar o envio de dados pela rede.

Aspectos operacionais envolvendo como as aplicações solicitam sua necessidade de QoS à rede, ou seja, como elas fazem suas reservas e como os elementos da rede devem proceder para garantir a qualidade de serviço solicitada, precisam ser definidos.

Através do protocolo de sinalização RSVP (*Resource ReSerVation Protocol)* [23], as aplicações solicitam suas necessidades de QoS à rede. Primeiramente a aplicação cliente identifica sua necessidade de QoS e solicita à rede, através do protocolo RSVP, a reserva de recursos que lhe é conveniente. A rede aceita eventualmente a solicitação e "tenta garantir" a reserva solicitada e com a reserva estabelecida, os fluxos de dados (*streams*) correspondentes à aplicação são identificados e roteados segundo a reserva feita para os mesmos.

O protocolo de sinalização RSVP atua sobre o tráfego de pacotes em redes (privadas ou públicas). É um protocolo eficiente que provê granularidade e controle fino das solicitações feitas pelas aplicações. Tem como principal desvantagem sua complexidade em relação a sua operação nos roteadores que, algumas vezes, pode conduzir a situações difíceis nos *backbones* de redes grandes, devido a grande troca de mensagens e necessidade de guardar estados dos fluxos nestes roteadores. É um protocolo bem aceito pelo mercado e está disponível na maioria dos sistemas operacionais (Unix, NT, Windows 98,...) e nos equipamentos de rede de vários fornecedores.

Em [26, 28, 68, 69] encontram-se os RFCs referente ao *IntServ*, que descreve a maneira como os elementos de rede devem proceder para oferecer a garantia da qualidade de serviço solicitada.

3.3.1.2 A Arquitetura de Serviços Diferenciados (*DiffServ*)

DiffServ [2] é uma arquitetura elaborada para evitar os problemas de escalabilidade e complexidade decorrentes da manutenção de informação de estado para cada fluxo e sinalização a cada nó na arquitetura *IntServ*. Nesta filosofia o tratamento do tráfego é feito pelos roteadores *DiffServ* de forma agregada. Os serviços diferenciados são obtidos através do campo DS (*Differentiated Service*) do cabeçalho IP, que retira do pacote informações do tipo de tratamento que o pacote deve obter em um determinado roteador (PHB – *Per Hop Behavior*). A complexidade é deixada para os roteadores de borda, enquanto que os roteadores do núcleo são mais simples. *DiffServ* [21] é uma outra iniciativa do IETF com o objetivo de permitir também o transporte de áudio, vídeo, dados em tempo real e dados "convencionais" na Internet.

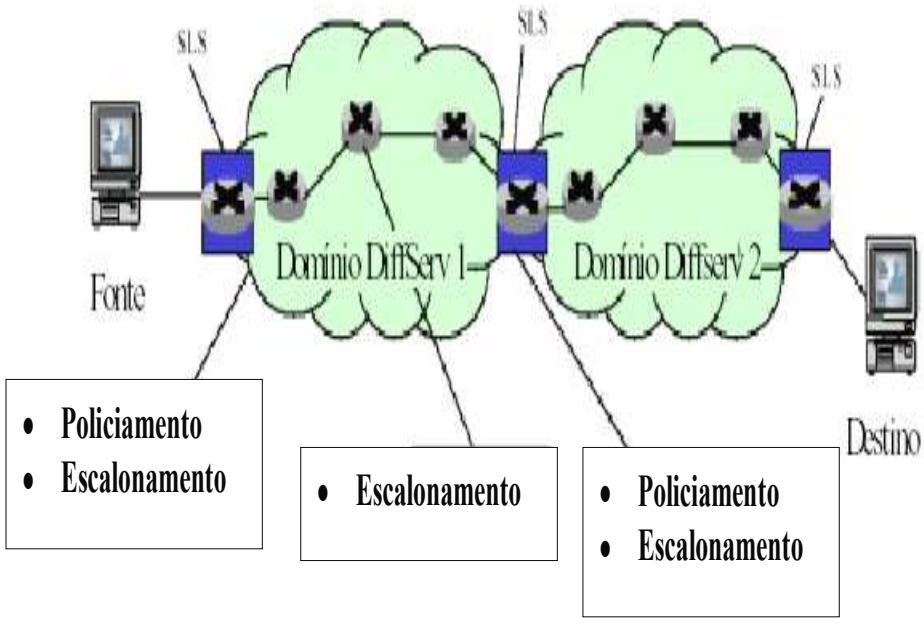


Figura 3-2 - Arquitetura de Serviços Diferenciados.

O princípio básico arquitetural de concepção do *DiffServ* é: A qualidade de serviço na solução *DiffServ* é garantida através de mecanismos de priorização de pacotes na rede.

Na solução *DiffServ* os pacotes são classificados, marcados e processados segundo a codificação rotulada no cabeçalho do pacote (DSCP - *Differentiated Service Code Point*) [22]. A idéia básica desta solução é reduzir o nível de processamento necessário nos roteadores para fluxos de dados (*streams*). Este objetivo é alcançado com a definição de poucas classes de serviços numa estrutura comum de rede.

Os fluxos gerados pelas aplicações (tráfego de pacotes IP) são agregados a classes de serviço em função da qualidade de serviço (QoS) especificada para cada fluxo. Esta é uma tarefa realizada nos roteadores de entrada do *backbone* (*Edge routers)* e, desta maneira, é simplificado o processamento nos roteadores intermediários *(Core*), além de ficar independente dos fluxos individuais das aplicações. Os roteadores intermediários roteiam "agregados de fluxos".

Na figura 3-3 são mostrados os blocos funcionais principais num equipamento roteador, utilizando a solução *DiffServ*. Todas as funções que aparecem neste diagrama, estão normalmente presentes nos roteadores de entrada e saída da rede (*Edge routers*) e, eventualmente, são acionadas nos roteadores intermediários (*Core routers/ backbone routers*).

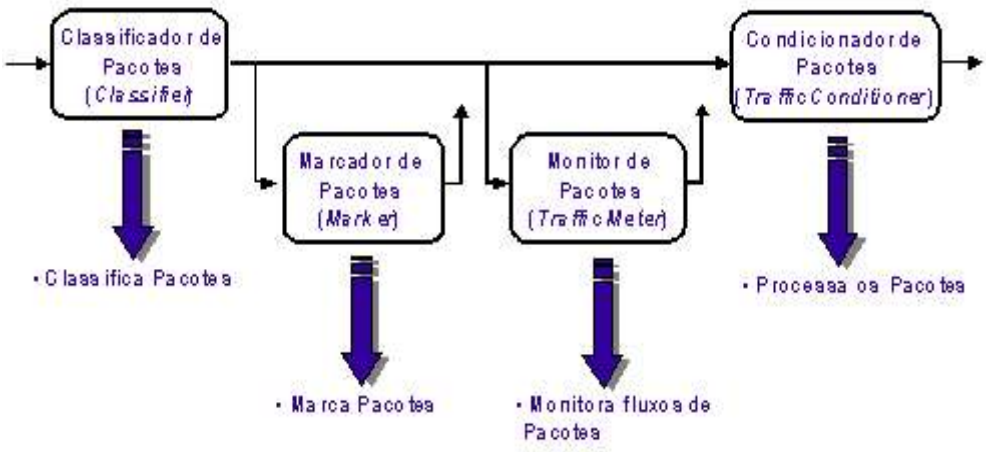


Figura 3-3 - Blocos Funcionais do DiffServ

A figura 3-4 ilustra a marcação dos DSCP para o protocolo IPv4. No IPv4 utiliza-se o campo TOS (*Type of Service*) do IP. No IPv6 utiliza-se o campo "*Traffic Class*".

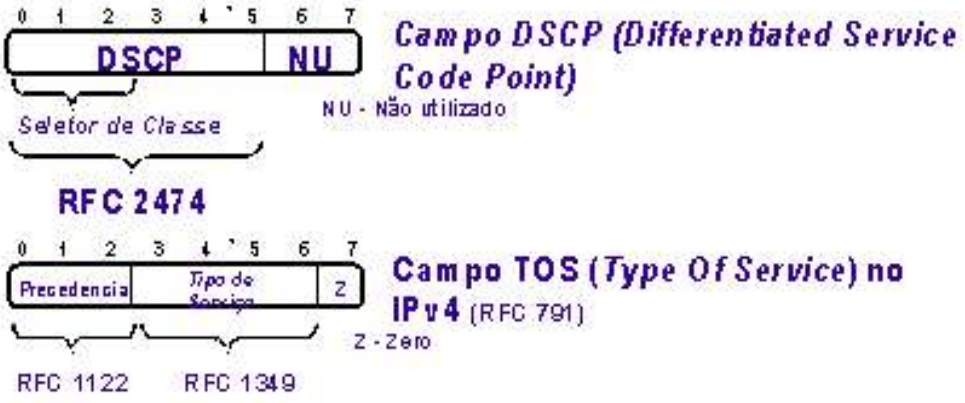


Figura 3-4 - DSCP - *Differentiated Service Code Point*

A operação do *DiffServ*, conforme [5], segue um roteiro. Cada pacote recebe um processamento baseado na sua marcação (DSCP). O *DiffServ* define duas classes de serviço que podem também ser entendidas como "comportamentos" (PHB - *Per-Hop Behavior)*, na medida em que definem como os equipamentos roteadores se comportam com relação aos pacotes (Como os pacotes são processados), estes comportamentos são:

Expedited Forwarding (EF) - Esta classe de serviço provê o maior nível de qualidade de serviço. A idéia é emular uma linha dedicada convencional minimizando os atrasos, probabilidade de perda e *jitter* para os pacotes. EF utiliza mecanismos de *traffic shaping,* buferização *(buffering*) e priorização de filas discutidos adiante.

Assured Forwarding (AF) - Esta classe de serviço emula um comportamento semelhante a uma rede com pouca carga mesmo durante a ocorrência de congestionamento. A latência negociada é garantida com um alto grau de probabilidade. O serviço AF define quatro níveis de prioridade de tráfego (Ouro, Prata, Bronze e *Best Effor*t) (Os níveis de prioridade foram inspirados na premiação dos jogos olímpicos) e para cada nível de prioridade

são definidas três preferências de descarte de pacotes. Este serviço usa mecanismos de *Traffic Shaping (Token Bucket*) e usa o algoritmo RED (*Randon Early Detection*), discutido adiante, durante o congestionamento.

Além dos serviços acima citados, outros serviços poderão ser definidos no escopo das recomendações *DiffServ*.

As soluções *IntServ* e *DiffServ* podem ser utilizadas conjuntamente e não são concorrentes ou mutuamente exclusivas. São soluções complementares. (RFC 2998 [74]). Como exemplo de utilização conjunta, poderíamos empregar a alternativa *DiffServ* no *backbone* de roteadores (*core*), uma vez que é uma solução mais "leve", e o *IntServ*/RSVP poderíamos empregar nas redes de acesso, isto devido as suas características que provê um bom controle com granularidade dos requisitos de QoS das aplicações.

Para uma relação completa de recomendações relacionadas com a arquitetura *DiffServ* consultar [21].

3.3.1.3 MultiProtocol Label Switching

MPLS (*Multiprotocol Label Switching*) é uma tecnologia emergente que, além de possibilitar um aumento no desempenho do encaminhamento de pacotes, facilita a implementação da Qualidade de Serviço (QoS), Engenharia de Tráfego (TE) e Redes Virtuais Privadas (VPN).

Os princípios da solução MPLS [24] são os seguintes:

Tecnicamente podemos afirmar que a solução MPLS tem algumas similaridades com a solução *DiffServ*, por exemplo: Os pacotes são marcados com um rótulo (MPLS Label) de 20 bits; os pacotes são marcados pelo MPLS na entrada da rede (MPLS *edge routers*) e os rótulos são retirados dos pacotes na saída da rede (MPLS *edge routers*).

O MPLS torna possível a interferência no roteamento da rede permitindo que seja adicionando novas rotas além das determinadas pelo roteamento IP padrão [29,30,31]. Estas rotas são conhecidas como Caminhos Comutados por Rótulos (LSPs - *Label SwitchedPaths*). O MPLS aumenta o controle sobre o roteamento pelos operadores da rede, uma característica importante para a

engenharia de tráfego e para a garantia de certos parâmetros de qualidade de serviços (QoS).

A indicação para onde o pacote deve ser encaminhado (*Forwarding*) na operação do MPLS é feito pela utilização de rótulos que direciona para próximo roteador (*Next hop*). Este aspecto operacional diferencia-o substancialmente das soluções anteriores na medida em que [32]:

O MPLS é uma solução mais orientada para uma engenharia de tráfego de pacotes na rede que para uma garantia efetiva de qualidade de serviço (QoS);

Um dos ganhos mais importantes com a utilização do MPLS é a simplificação da função de roteamento nos roteadores, reduzindo assim a carga de roteamento nestes roteadores e as suas latências.

A redução da latência nos roteadores conduz a uma melhoria nas condições de operação na rede e, conseqüentemente, a qualidade de serviço. Mas, o MPLS não possui mecanismos de controles específicos que garanta QoS na rede. Uma característica importante do MPLS é sua independência de protocolo da rede, esta é uma transparência significativa. O MPLS é residente apenas nos roteadores. O MPLS não é controlável pela aplicação, isto é, não há API (*Application Programmming Interface*) disponível para o MPLS nos *hosts*.

O MPLS tem sua operação efetivada com a distribuição de rótulos (MPLS *Labels*) entre os roteadores e com a gerência dos mesmos. O protocolo de sinalização LDP (*Label Distribution Protocol*) [25] foi desenvolvido para executar esta função.

3.3.1.4 Roteamento baseado em QoS (QoSR)

O roteamento baseado em QoS (QoS *Routing*) [4] é um mecanismo de roteamento que, baseado no conhecimento da disponibilidade de recursos da rede, bem como nos requisitos de QoS dos fluxos como largura de banda e atraso, seleciona o caminho percorrido pelos pacotes de um fluxo. Três maneiras o QoSR utilizará para estender o paradigma de roteamento na Internet [26]. A primeira maneira é: para suportar tráfego baseado em novos serviços, proporcionados por *IntServ* ou *DiffServ*, devem ser encontrados caminhos múltiplos entre roteadores, cada um com as suas características de QoS. Alguns desses serviços poderão necessitar da distribuição de mais do que uma métrica, como largura de banda e atraso. A segunda maneira trata do roteamento utilizado atualmente na Internet, este tratamento direciona o tráfego para uma nova rota imediatamente quando encontra um caminho "melhor". Assim, o tráfego é redirecionado para o novo caminho, mesmo que o antigo esteja apto a satisfazer as necessidades dos fluxos. Isto, além de gerar grande instabilidade, aumenta a variação do atraso dos fluxos, as escolhas de roteamento muitas vezes são inadequadas. A terceira maneira envolve os caminhos alternativos que não são suportados, mesmo que possam atender às características de várias aplicações. Por este motivo, a alternativa QoSR pode encontrar um caminho que pode ser mais longo, mas que, com certeza, muito menos sobrecarregado. Caminho mais curto geralmente é o mais congestionando. O tráfego na rede, utilizando esta técnica, pode ser distribuído de maneira mais equilibrado. É importante deixar claro a diferença entre QoSR e reserva de recursos. Os protocolos de reserva de recursos, como o RSVP, não proporcionam nenhum mecanismo para encontrar um caminho que tenha recursos suficientes para satisfazer os níveis de QoS requisitados [60], eles, apenas, oferecem um método para requisitar e reservar recursos da rede. QoSR não inclui um mecanismo para reservar os recursos necessários, mas permite que seja determinado um caminho com uma grande probabilidade de satisfazer

a requisição de QoS. Desta maneira, as duas técnicas apresentam características complementares e geralmente são implementadas em conjunto. É uma combinação que permite exercer um controle significativo sobre rotas e recursos, ao custo de informações adicionais de estado e tempo de configuração.

Uma das grandes diferenças entre QoSR e o roteamento convencional é a manutenção de estado sobre a capacidade dos recursos da rede em atender requisitos de QoS. Os seus principais objetivos são [26]:

- Determinação dinâmica de possíveis caminhos: Embora QoSR possa encontrar um caminho que atenda os requisitos de QoS de um fluxo, o direcionamento do tráfego para ele pode depender de outras restrições (Roteamento Baseado em Restrições).
- Otimização da utilização dos recursos: Um esquema de QoSR pode auxiliar na utilização eficiente dos recursos da rede, aumentando a vazão total alcançada pela rede. Caminhos ociosos podem ser encontrados para satisfazer demandas por requisitos específicos de QoS, o que é impossível com o roteamento convencional. Isso pode ser usado para realizar Engenharia de Tráfego.
- Degradação graciosa de desempenho: O roteamento dependente de estado pode compensar problemas transientes na rede, escolhendo caminhos alternativos, que permitem que as aplicações melhor se adaptem as condições momentâneas da rede.

O cálculo de rotas com base em restrições de QoS é um processo mais caro em relação ao atual paradigma de roteamento da Internet. O importante é analisar e determinar o ponto ótimo onde a melhoria em desempenho pela adoção de QoSR vale a pena comparado com o aumento nos custos. Este custo adicional originado pelo QoSR tem dois componentes principais [27]: custo computacional e sobrecarga de protocolo. O custo computacional é devido ao algoritmo mais sofisticado para escolha dos caminhos e à necessidade de processá-lo mais freqüentemente.

Em geral, existem soluções polinomiais que envolvem o uso de largura de banda e alguma outra métrica (atraso, variação do atraso, confiabilidade), mas assim mesmo o processamento é complexo. No entanto, os avanços tecnológicos permitem que se diminua a importância do custo computacional pela utilização de processadores mais velozes e memórias maiores.

3.3.1.5 Dimensionamento

Esta é uma alternativa bastante simples. Nesta alternativa, o projeto da rede dimensiona seus recursos de forma que não exista congestionamento. Por exemplo, dimensiona-se uma grande largura de banda para que não exista congestionamento ou de períodos de escassez de recursos durante a operação da rede.

Esta é uma solução que apresenta duas dificuldades principais. A primeira está relacionada ao custo associado na implementação do super dimensionamento. Esta dificuldade muitas vezes é inviável para as redes WAN. A segunda dificuldade é que devido à multiplicidade e diversidade de equipamentos utilizados e a própria complexidade das redes, torna-se difícil a identificação dos pontos de ocorrência de congestionamento da rede. Esta solução, embora implementável, não é uma solução realista.

3.4 Requisitos Essenciais em Redes com QOS

Conforme [32], o conjunto de requisitos considerados essenciais para implementação dos mecanismos de QoS em redes são: Vazão e Largura de Banda, Atraso Fim-a-Fim (Latência), Variação do Atraso (*jitter*), Perda de

Pacotes e disponibilidade. Estes requisitos são utilizados na especificação das SLAs definindo os parâmetros de qualidade de serviço.

3.4.1 Largura de Banda

Conforme dicionário de Telecomunicações e Informática, a largura de uma banda de freqüência eletromagnética significa quão rápido os dados fluem, seja numa linha de comunicação ou no barramento de um computador. Quanto maior a largura de banda, mais informações podem ser enviadas num dado intervalo de tempo. Pode ser expressa em bits por segundo (bps), *bytes* por segundo (Bps), Kilobits por segundo (Kbps), Megabits por segundo (Mbps) ou ciclos por segundo (Hz).

A banda indica a capacidade máxima de transmissão teórica de uma conexão, mas na medida em que a utilização se aproxima da largura de banda teórica máxima, fatores negativos como atraso de transmissão podem causar deterioração em qualidade.

3.4.2 Vazão

A vazão é o montante de tráfego de dados movidos de um nó de rede para outro em um dado período de tempo, também expresso em Kbps ou Mbps [33]. A vazão também pode ser expressa em pacotes por segundo (pps).

A vazão (banda) é um dos parâmetros básicos de QoS, sendo necessário para a operação adequada de qualquer aplicação. A tabela 3-1 ilustra a vazão típica de algumas aplicações:

Aplicação	
Aplicações Transacionais	**1 Kbps a 50 Kbps**
Voz	**10 Kbps a 120 Kbps**
Aplicações Web (WWW)	**10 Kbps a 500 Kbps**
Transferência de Arquivos (Grandes)	**10 Kbps a 1 Mbps**
Video (*Streaming*)	**100 Kbps a 1 Mbps**
Aplicação Conferência	**500 Kbps a 1 Mbps**
Vídeo MPEG	**1 Mbps a 10 Mbps**
Aplicação Imagens Médicas	**10 Mbps a 100 Mbps**
Aplicação Realidade Virtual	**80 Mbps a 150 Mbps**

Tabela 3-1 - Vazão Típica de Aplicações em Rede

3.4.3 Atraso Fim-a-Fim (Latência)

Latência é o tempo envolvido entre o envio de uma mensagem a partir de um nó e a recepção desta mensagem no nó destino. Pode ser definido, também, como o atraso de uma ligação de telefonia IP. Ou seja, o tempo decorrido entre o momento em que uma pessoa fale (transmissão) e o instante em que o ouvinte do outro lado da linha escuta o sinal de voz (recepção).

Em um roteador, é tratado como o tempo decorrido entre recepção de um pacote e sua transmissão, também chamado de tempo de propagação.

Na prática, duas maneiras existem para medir a Latência entre dois pontos de uma rede TCP/IP: o atraso de ida (*one-way delay*) e o atraso de ida e volta *(round-trip delay*)[34]. O atraso de ida é o tempo medido entre o envio de uma mensagem a partir de um nó e a recepção desta mensagem pelo nó destino. Este atraso acontece devido ao caminho de transmissão ou em um dispositivo presente na trajetória de transmissão. Em outras palavras o atraso é o tempo necessário para um pacote percorrer a rede, medido do momento em que é transmitido pelo emissor até ser recebido pelo receptor [18]. O atraso de ida e volta considera, evidentemente, os tempos nos dois sentidos. Na medida

do atraso de ida e volta é necessário a utilização de dois pacotes IP diferentes, o primeiro pacote (IP1) é gerado pelo emissor, e o segundo pacote (IP2) é enviado pelo receptor como resposta à requisição do primeiro pacote (IP1) recebido.

De maneira geral, a latência é considerada como o somatório dos atrasos sofrido pela mensagem através da rede e dos equipamentos utilizados na comunicação. Em termos de aplicação, a latência (atrasos) se traduz em um tempo de resposta (tempo de entrega da informação - pacotes) para a aplicação.

A latência sofre influência de vários fatores em uma rede, sendo os principais fatores os seguintes:

- Atraso de propagação (*Propagation Delay*) – este atraso é o tempo gasto para a propagação do sinal elétrico ou propagação do sinal óptico no meio que está sendo utilizado (fibras ópticas, satélite, coaxial), este é um parâmetro característico do meio e é imutável.
- Velocidade de transmissão – este parâmetro pode ser controlado pelo gerente e visa a adequação da rede à qualidade de serviço solicitada. Nas redes locais (LANs), as velocidades de transmissão são normalmente elevadas, podendo ser superior a 10 Mbps dedicada por usuário (ex: utilizando LAN Switches). Quando envolve redes de longa distância (Redes corporativas estaduais e nacionais, redes metropolitanas, intranets metropolitanas) as velocidades de transmissão são dependentes da tecnologia de rede WAN escolhida (Linhas privadas, *Frame Relay*, satélite, ATM). Embora exista escolha da velocidade adequada para garantia da qualidade de serviço, observam-

se restrições e/ ou limitações nas velocidades contratadas, isto devido aos custos mensais envolvidos na operação da rede. Existem, também, algumas restrições quanto à disponibilidade tanto da tecnologia quanto da velocidade de transmissão desejada. De uma maneira geral, as WANs trabalham com vazões da ordem de Kilobits por segundo (Kbps) ou de alguns megabits por segundo (Mbps) para grupos de usuários.

- Processamento nos equipamentos - este fator contribui para a latência da rede e refere-se ao atraso sofrido no processamento realizado nos equipamentos. Podemos exemplificar este atraso, verificando o processamento dos dados numa rede IP ao longo do percurso entre origem e destino. Neste percurso encontramos as seguintes fontes de atrasos:
  - Roteadores (comutação de pacotes)
  - LAN Switches (comutação de quadros)
  - Servidores de Acesso Remoto (RAS) (comutação de pacotes)
  - *Firewalls* (processamento no nível de pacotes ou no nível de aplicação)

Como a latência é um parâmetro fim-a-fim, existe uma contribuição dos equipamentos finais (*hosts*). Nestes equipamentos, temos que considerar:

- Capacidade de processamento do processador;
- Disponibilidade de memória;
- Mecanismos de *cache*;
- Processamento nas camadas de nível superior da rede (Programa

de aplicação, camadas acima da camada IP).

Podemos observar que os *hosts* são também um fator importante para a qualidade de serviço e, em determinados casos, podem ser um ponto crítico na garantia de QoS. Esta consideração é particularmente válida para equipamentos servidores (*Servers*) que têm a tarefa de atender solicitações simultâneas de clientes em rede.

3.4.4 jitter

Segundo dicionário de Telecomunicações e Informática, *jitter* é o fenômeno caracterizado pelo desvio no tempo ou na fase de um sinal de transmissão de pacotes de dados. Pode ser responsável por erros e perda de sincronismo em comunicações assíncronas em altas velocidades, por exemplo, em telefonia IP. A variação no tempo de chegada de pacotes prejudica a qualidade da conversão – se um pacote não chega a tempo de se encaixar em seu lugar no fluxo de dados, repete-se o pacote anterior. Pode ser corrigido com a adoção de uma memória adicional (*jitter* buffer).

A importância do parâmetro *jitter* para a qualidade de serviço é observada na execução das aplicações em rede cuja operação adequada dependa de alguma forma da garantia de que as informações (pacotes) devam ser processadas em períodos de tempo bem definidos. Este caso se aplica, por exemplo, a aplicações de tempo real e a aplicações de voz e fax sobre IP (VoIP), entre outras.

Para uma rede de computador, o *jitter* é entendido como a variação no tempo e na seqüência de entrega das informações *(Packet-Delay Variation)* devido à variação na latência (atrasos) da rede.

A contribuição dos atrasos que a rede e seus equipamentos impõem à informação é variável devido a uma série de fatores, a saber:

- Tempos de processamento diferentes nos equipamentos intermediários (roteadores, switches,...);

•Tempos de retenção diferentes impostos pelas redes públicas (*Frame relay*, ATM, X.25, IP,...) e

•Outros fatores ligados à operação da rede.

A figura 3-5 mostra graficamente o efeito do *jitter* entre a entrega de pacotes na origem e o seu processamento no destino. É importante observar que o *jitter* não causa somente uma entrega de pacotes com periodicidade variável *(Packet-Delay Variation)* como também pode entregar os pacotes fora de ordem.

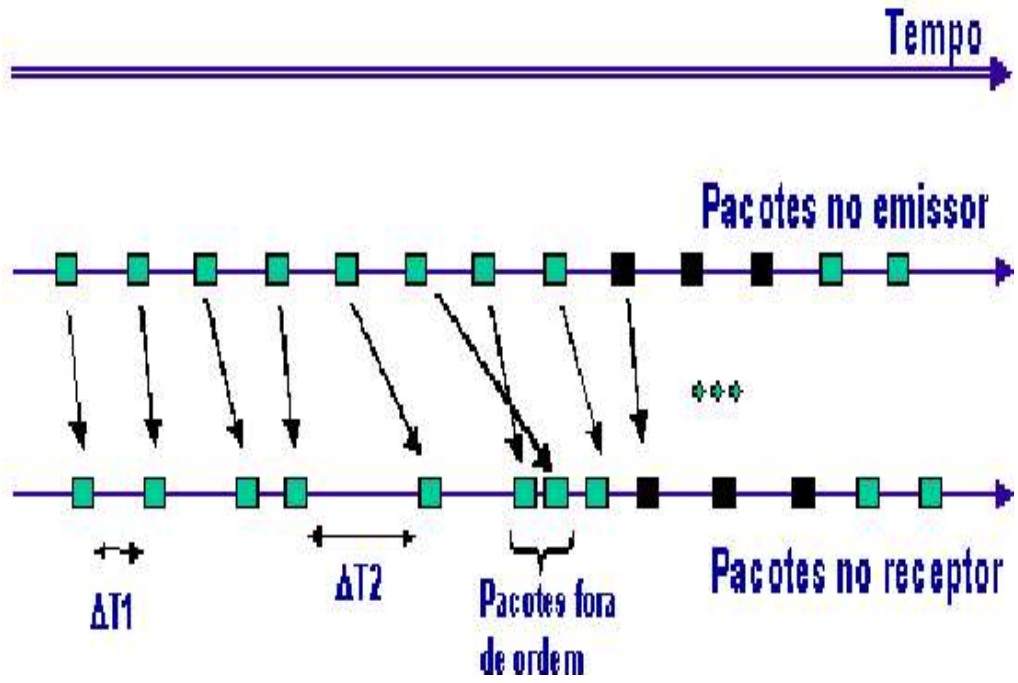


Figura 3-5 - Efeito do *jitter* para as Aplicações

Pacotes fora de ordem é um problema que o protocolo de transporte como o TCP (*Transmission Control Protocol)* poderia resolver, pois ele verifica o sequenciamento das mensagens e faz as devidas correções. No entanto, as aplicações multimídia geralmente optam por utilizar o protocolo UDP (*User Datagram Protocol*) ao invés do TCP pela maior simplicidade e menor *overhead* deste protocolo. Nestes casos, o problema de sequenciamento deve ser resolvido por protocolos de mais alto nível normalmente incorporados

à aplicação como, por exemplo, o RTP (*Real Time Transfer Protocol*).

O *jitter* introduz distorção no processamento da informação na recepção e deve ter mecanismos específicos de compensação e controle que dependem da aplicação em questão. Genericamente, uma das soluções mais comuns para o problema consiste na utilização de *buffers*.

3.4.5 Perdas

Em redes IP, perdas de pacotes ocorrem principalmente quando há descarte de pacotes nos roteadores e *switch routers* (erros, congestionamento) e devido a erros ocorridos na camada 2 (PPP - *Point-to-Point Protocol, Ethernet, Frame Relay*, ATM) durante o transporte dos mesmos. Estas perdas são geralmente, problemas sérios para determinadas aplicações como, voz sobre IP, onde o efeito da perda de pacotes em trechos de voz digitalizada implica numa perda de qualidade, que eventualmente, não são aceitáveis para a aplicação. A preocupação com perdas e o tratamento dado a estas perdas é uma questão específica de cada aplicação.

A qualidade de serviço da rede (QoS) tem a preocupação de especificar e garantir taxas de perdas dentro de limites especificados que permitam que a aplicação tenha uma operação adequada.

3.4.6 Disponibilidade

Em qualidade de serviço, disponibilidade se traduz como a medida da garantia de execução da aplicação ao longo do tempo e é abordada normalmente na fase de projeto da rede. Disponibilidade depende de fatores como a disponibilidade dos equipamentos utilizados na rede proprietária (Rede do cliente) (LAN, MAN ou WAN) e a disponibilidade da rede pública, quando esta é utilizada (Operadoras de telecomunicações, *carriers, ISPs - Internet Service Providers*).

Os negócios da rede para se tornarem viáveis necessitam que o parâmetro disponibilidade seja bastante alto, por isso, a disponibilidade é um requisito bastante rígido. Por exemplo, requisitos de disponibilidade acima de 99% do tempo são comuns para QoS de aplicações WEB, aplicações

cliente/servidor e aplicações de forte interação com o público, dentre outras, assim como para a viabilização de negócios (Comércio eletrônico, *home-banking*, atendimento *online*, transações *online*).

3.5 Considerações sobre Este Capítulo

Este capítulo mostrou as definições e conceitos relacionados à Qualidade de Serviços, como QoS é provido através de uma rede, os avanços alcançados por esta tecnologia e as alternativas técnicas mais usadas para provimento de QoS. Também enumerou e conceituou os requisitos essenciais presentes em uma rede com QoS. O capítulo seguinte enfoca os mecanismos que implementam QoS em uma rede. Neste sentido são mostrados os algoritmos de prioridade, de escalonamento, de controle de filas e de controle de congestionamento e como eles são utilizados para a implementação das políticas de QoS.

4 QOS - MECANISMOS

As alternativas técnicas para provimento de QoS em redes de computadores são implementadas através da utilização de diversos tipos de mecanismos, chamados de mecanismos de QoS. Estes mecanismos compreendem Classificação e Marcação, Gerenciamento de Congestionamento, Controle de Congestionamento, Policiamento e Moldagem de Tráfego, e Controle de Admissão. Para que estes mecanismos funcionem é necessário que os Algoritmos de prioridade, Algoritmos de escalonamento, Algoritmos de controle de filas e os Algoritmos de congestionamento sejam implementados conforme política adotada para solução do problema específico. Descreveremos a seguir cada mecanismos com suas características e funções.

4.1 Classificação e Marcação

A classificação disponibiliza um serviço preferencial a um determinado tipo de tráfego. O fluxo de pacotes é identificado, depois ele pode ou não ser marcado. A classificação usa o descritor de tráfego para categorizar um pacote dentro de um grupo específico e o torna acessível aos mecanismos de QoS existente na rede. Usando classificação de pacotes, o tráfego de uma rede pode ser particionado dentro de múltiplos níveis de prioridades ou classes de serviços.

Após a etapa de classificação, os outros mecanismos de QoS podem manipular estes pacotes classificados para fazer o controle de congestionamento, alocação de banda, e outros controles. As redes podem ser configuradas para aceitar fluxos classificados ou para fazerem reclassificações conforme política estabelecida. A classificação pode ser feita com base em portas físicas, IP fonte e destino ou endereço MAC, porta de aplicação, tipo de protocolo, e outros critérios que utilizam listas de acessos.

4.1.1 Precedência IP e DSCP

A classificação baseia-se em um campo do cabeçalho IP, denominado *DS-field*. Este campo é uma redefinição do campo *Type of Service* presente no cabeçalho IPv4, ou *Class of Traffic* do IPv6. Este campo é composto de oito bits, sendo que os seis primeiros bits do *DS-field* são usados para identificar o valor de DSCP (*differentiated services codepoint*), usado na seleção do PHB (*per-hop-behavior)*. Os dois últimos bits (CU - *currrently unused*), não são utilizados atualmente, e por isso, desconsiderados pelos mecanismos de QoS.

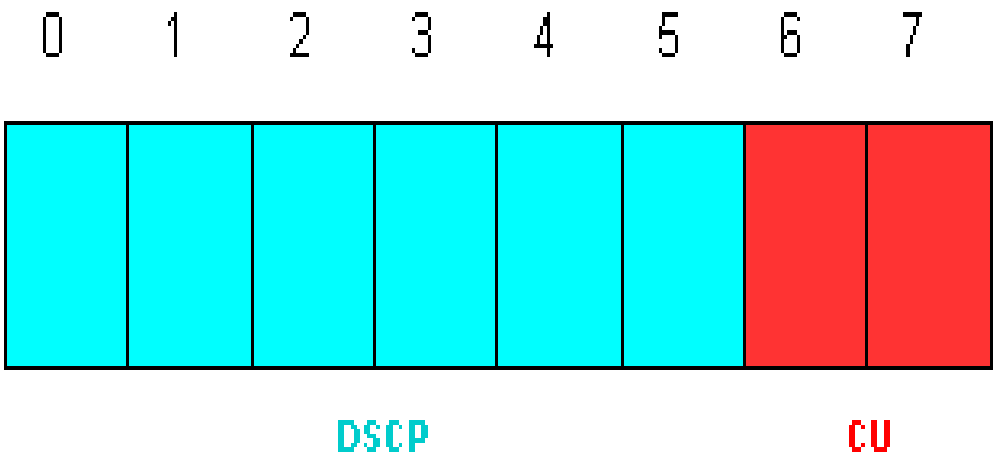


Figura 4-1 - Representação do *DS-field*

PHBs são comportamentos individuais aplicados em cada roteador. Existem dois PHBs padronizados, que são:

•*Expedited Forwarding* PHB (RFC 2598) [29].

Este PHB pode ser usado para construir serviço fim-a-fim com as seguintes características:

•Baixa perda;

•Baixa latência;

•Baixa variação de atraso e

•Banda assegurada.

Do ponto de vista do nó final, tal serviço aparenta ser uma conexão ponto-a-ponto ou uma *virtual leased line*. Este serviço também tem sido descrito como serviço *Premium*.

•*Assured Forwarding* PHB (RFC 2597) [36].

O grupo AF PHB fornece entrega de pacotes IP em 4 classes onde, para cada uma, é alocada uma determinada quantidade de recursos para o encaminhamento do tráfego (*buffer* e banda). Dentro de cada classe AF, um pacote IP pode ser associado a um dentre três níveis diferentes de precedência de descarte. Este PHB pode ser usado para implementar serviços que garantam um valor de banda, mesmo em tempos de congestionamento. Um exemplo é o serviço Olímpico (*Gold, Silver e Bronze*). A tabela 4-1 mostra valores de DSCP recomendados para identificar os PHBs.

Default PHB Codepoint (melhor esforço)	000000 (0)			
EF PHB	101110 (46)			
AF PHB	Classes	Low Drop Prec	Prec	High Drop Prec
	Class 1	001010 (10)	001100 (12)	001110 (14)
	Class 2	010010 (18)	010100 (20)	010110 (22)
	Class 3	011010 (26)	011100 (28)	011110 (30)
	Class 4	100010 (34)	100100 (36)	100110 (38)

Tabela 4-1 – Valores de DSCP recomendados para identificar os PHBs

Conforme descrito acima, para se obter, no protocolo da Internet (Ipv4) [7], as classes de serviços (*Class of Service* - CoS), são utilizados 3 bits do campo ToS (*Type of Service*) do cabeçalho IP, conforme ilustra a figura 4-2 abaixo. Este campo foi concebido e reservado para indicar os tipos de serviços, mas nunca havia sido utilizado em nenhuma implementação. A classificação de pacotes dá funcionalidades para precedência IP (*IP Precedence)*.

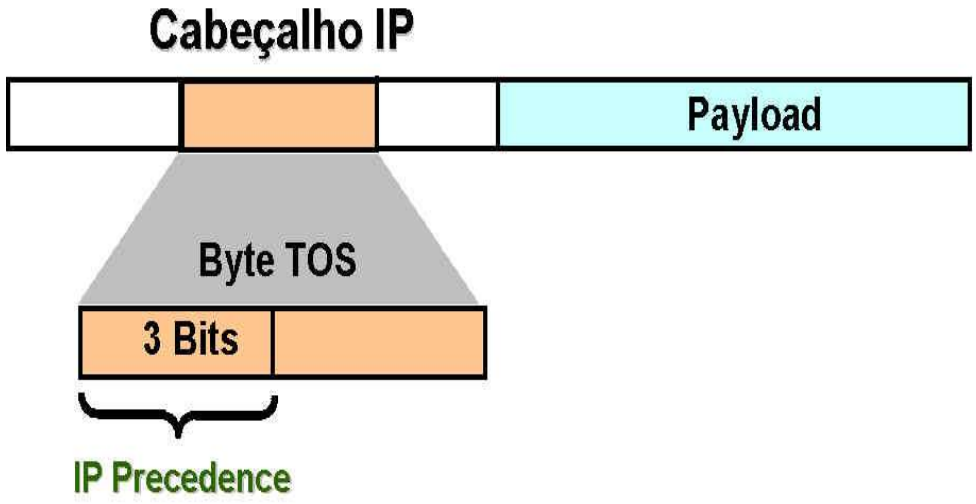


Figura 4-2 - Precedência IP no Campo ToS

Os 3 bits para o campo Precedência IP oferece as seguintes configurações e funções:

- 0 - *rotine*: estabelece a precedência rotina;
- 1 - *priority*: estabelece a precedência prioridade;
- 2 - *immediate*: estabelece a precedência prioridade;
- 3 - *flash*: estabelece a precedência *flash*;
- 4 - *flash-override*: estabelece a precedência *flash-override*;
- 5 - *critical*: estabelece a precedência crítica;
- 6 - *internet*: estabelece a precedência *internetwork control*;

•7 - *network*: estabelece a precedência controle de rede.

A prioridade no tratamento e alocação de recursos da rede está relacionada com o nível de classificação do pacote, quanto maior o nível de classificação do pacote maior será a prioridade . Não é possível habilitar um pacote com nível 6 ou 7, pois estes níveis são reservados para as aplicações de controle e gerência da rede, também não é possível modificar um pacote já marcado. Todos os pacotes são normalmente marcados com o nível zero.

Associando um pacote com um *Ip Precedence* ou IP DSCP, permitimos aos usuários classificar o tráfego baseados nos valores de *Ip Precedence* e IP DSCP, dependo do valor que esteja marcado. Isto feito serão usados para decidir como os pacotes serão tratados por outros mecanismos de QoS objetivando controlar tráfego.

O IP DSCP, com já descrito, corresponde aos seis primeiros bits no *byte* ToS, enquanto o IP *Precedence* corresponde aos primeiros três bits. O *Ip Precedence* é atualmente parte do IP DSCP. Desta maneira, ambos os valores não podem ser setados simultaneamente. Se isto ocorrer o valor IP DSCP será escolhido.

4.2 Gerenciamento de Congestionamento

O controle de congestionamento é efetivado na medida que o congestionamento se estabelece em uma comunicação de rede. Para ordenar o *overflow* de tráfego que chega, os elementos de rede usam os algoritmos de enfileiramento. Estes algoritmos determinam alguns métodos de priorização para o fluxo de saída. Cada algoritmo foi designado para resolver um problema específico de tráfego e tem efeitos sobre o desempenho da rede. Há vários mecanismos de enfileiramento para controle e prevenção de congestionamento em interfaces de roteadores (*Etherne*t, seriais, *Frame Relay*, etc.) e *switches* nível 3, aplicáveis tanto em redes WAN como em LAN. As principais são apresentadas a seguir.

4.2.1 **FIFO - *first in first out***

Este tipo de enfileiramento, geralmente é implementado para fazer o controle de tráfego nas conexões seriais dos roteadores. A fila FIFO (o primeiro a entrar é o primeiro a sair) é um mecanismo de armazenamento e repasse (*store and forward*) que não implementa nenhum tipo de classificação.

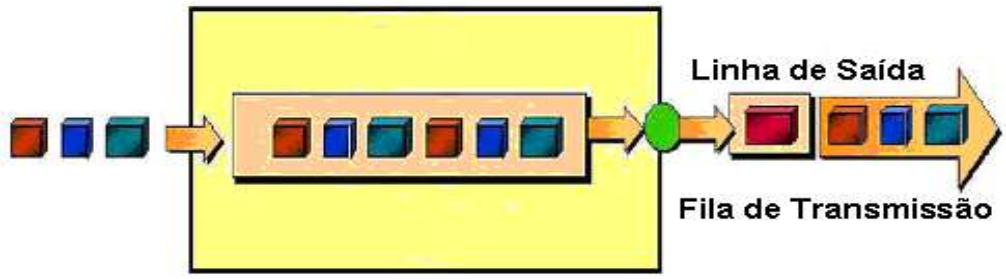


Figura 4-3 - Operação da Fila FIFO

A alocação da banda é determinada pela ordem de chegada dos pacotes. O pacote que chega primeiro é logo atendido. Hoje, este é o tratamento *default* da fila nos roteadores, já que não requer nenhuma configuração. O tráfego de rajada causa problemas neste mecanismo, pois pode causar longos atrasos em aplicações sensíveis ao tempo. Por esta situação, as filas FIFO não são uma boa indicação para aplicações que requerem QoS.

4.2.2 **Enfileiramento *Class Based Queueing (CBQ)***

A característica do CBQ [35] é criar classes de uma forma hierárquica onde é reservado um certo percentual de banda para cada uma. Ele dispõe de dois escalonadores com o objetivo de garantir aos fluxos de tempo real baixos valores de atraso nas filas de espera e ao mesmo tempo a não monopolização da ligação por parte destes fluxos. O CBQ embute a possibilidade de classes utilizarem um percentual de largura de banda maior do que o fornecido a elas, desde que não prejudiquem o fluxo de outras classes. Isto possibilita um maior aproveitamento da ligação. Assim, ele trabalha no sentido de dar prioridade às aplicações mais exigentes, ao fornecer maior percentual de banda, evitando que aplicações menos favorecidas entrem numa situação de postergação indefinida. Para entender melhor a estrutura hierárquica do CBQ, um exemplo

é mostrado na figura 4-4. Nesta figura é mostrado uma estrutura hierárquica na qual as agências recebem um certo percentual da banda e o distribui entre as suas folhas que são aplicações com requisitos distintos. No canto superior esquerdo, é indicado o seu nível, no canto inferior esquerdo a sua prioridade e os percentuais indicam a quantidade de largura de banda fornecida a cada classe.

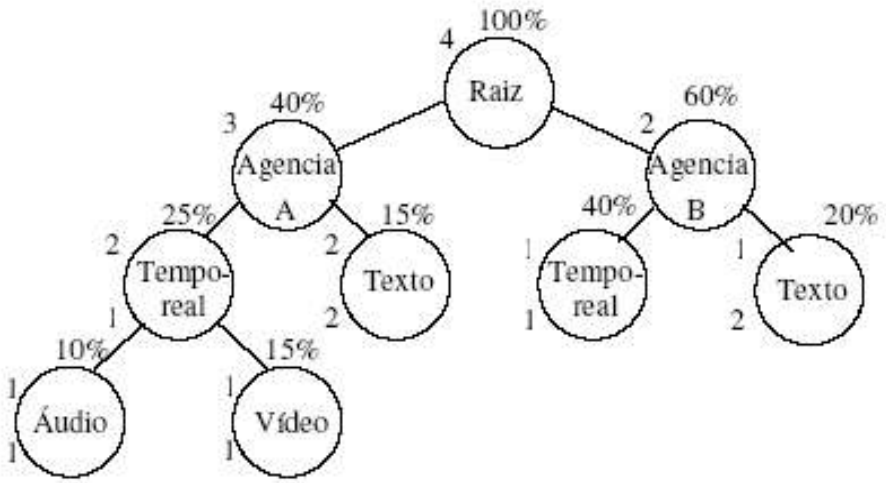


Figura 4-4 - Exemplo de estrutura hierárquica do CBQ

4.2.3 **Enfileiramento *Fair Queueing (FQ) e Weighted Fair Queueing (WFQ)***

Este algoritmo se propõe a oferecer uma alocação mais justa da banda entre os fluxos de dados. Neste algoritmo de enfileiramento, há uma ordenação de mensagens em sessões, sendo alocado para cada sessão, canal (figura 4-5), uma fração da banda. O último bit que atravessa o canal é a referencia para ordenação da fila.

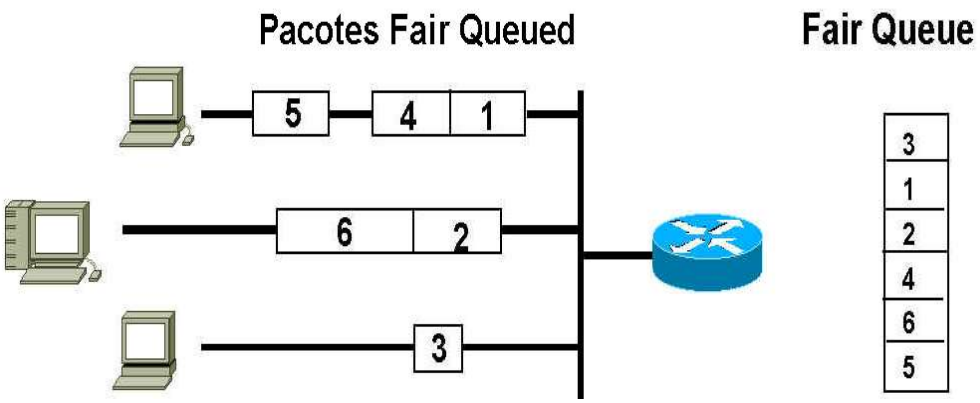
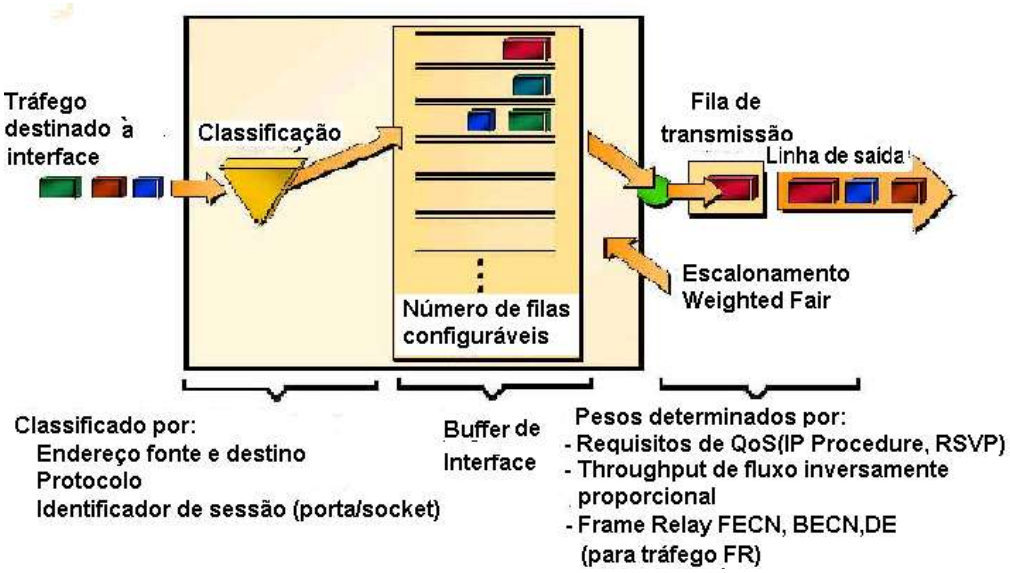


Figura 4-5- Filas *Fair Queueing*



Uma variação do FQ é o algoritmo WFQ - *Weighted Fair Queueing,* este algoritmo é uma implementação Cisco que tem como princípio ponderar determinados tipos de fluxo. O algoritmo coloca o tráfego prioritário (interativo) para frente da fila para reduzir o tempo de resposta e, de forma justa, compartilha o restante da banda com os outros tipos de fluxo. Este algoritmo se adapta automaticamente às variações do tráfego, sendo bastante útil em conexões seriais de baixa velocidade até 2 Mbps. A figura 4-6 mostra a operação do Algoritmo WFQ relevando suas principais características.

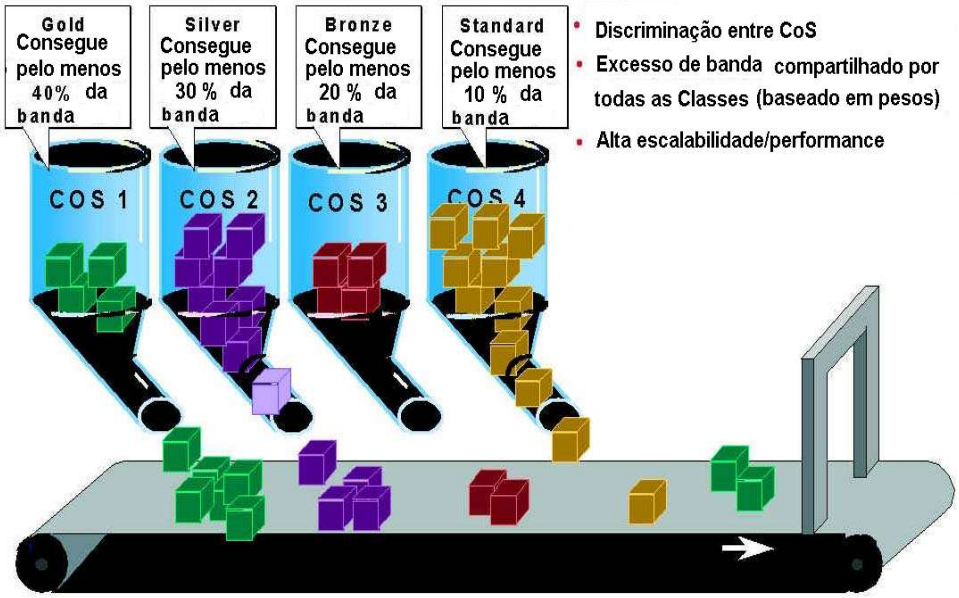


Figura 4-6 - Operação do Algoritmo WFQ

Analisando a figura 4-7 abaixo, podemos verificar que a classificação dos fluxos de dados pode ser realizada por endereço fonte ou destino, por protocolo, pelo campo precedência IP, pelo par porta/*socket*, ou outros parâmetros. É possível configurar o número de filas e a ponderação pode ser estabelecida por precedência IP. Conforme [7], outros protocolos de QoS como o RSVP, o tráfego *Frame Relay*, o VoFR (*Voice over Frame Relay*) através dos parâmetros FECN (*Forward Explicit Congestion Notification*), BECN (*Backward Explicit Congestion Notification*) e DE (*Discard Eligible*), podem ser usados em conjunto com a precedência IP para fazer a ponderação.

Figura 4-7 - Filas WFQ

4.2.4 **Enfileiramento *Class Based Weighted Fair Queueing* (CBWFQ)**

CBWFQ estende a funcionalidade padrão de WFQ para fornecer a sustentação para classes de tráfego. Para CBWFQ, você define as classes do tráfego baseadas em critérios de conformidade presente nos protocolos, listas do controle de acesso (ACLs), e interfaces de entrada. Os pacotes que estão em conformidade com os critérios estabelecidos para uma classe constituem o tráfego para essa classe. Uma fila FIFO é reservada para cada classe, e o

tráfego que pertence a uma determinada classe é dirigido à fila reservada para esta classe.

Uma vez que uma classe foi definida de acordo com seus critérios de conformidade, pode-se configurar suas características. Para caracterizar uma classe, deve-se lhe atribuir a largura de banda, o peso, e o limite máximo de pacotes. A largura de banda atribuída a uma classe é a largura de banda garantida a esta classe durante o congestionamento.

Depois que uma fila alcança seu limite configurado, enfileiramento de pacotes adicionais à classe faz com que a tail drop ou a packet drop aconteçam, dependendo de como a política da classe foi configurada.

4.2.5 **Enfileiramento *Priority Queueing* (PQ)**

O enfileiramento *Priority Queueing* - PQ (enfileiramento prioritário), classifica o tráfego de entrada em quatro níveis de prioridade (figura 4-8): alta, média, normal e baixa (*high, medium, normal* e *low*). Os pacotes não classificados são marcados, por *default*, como normal.

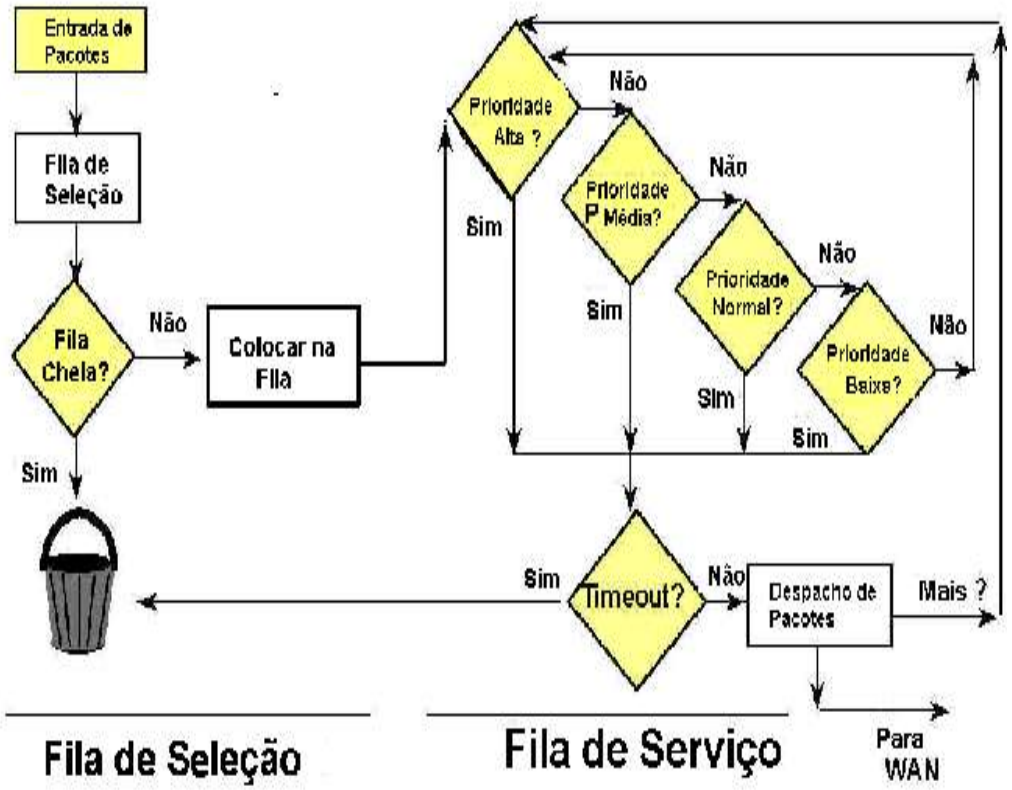


Figura 4-8 - Operação do Enfileiramento *Priority Queueuing*

Uma vez classificado e marcado como prioritário, o tráfego tem preferência absoluta durante a transmissão. Por este motivo, este método requer muito cuidado na sua utilização, para evitar que aplicações de menor prioridade tenham longos atrasos e aumento de *jitter*. O tráfego de menor prioridade pode até nunca ser transmitido, se o de maior prioridade tomar toda a banda. A conexão de baixa velocidade tem probabilidade maior deste problema ocorrer. A fila *default* sempre tem que ser habilitada para comportar o fluxo não classificado. Caso contrário, todo este fluxo não classificado(sem uma correspondente lista de prioridade) também poderá não ser enviado.

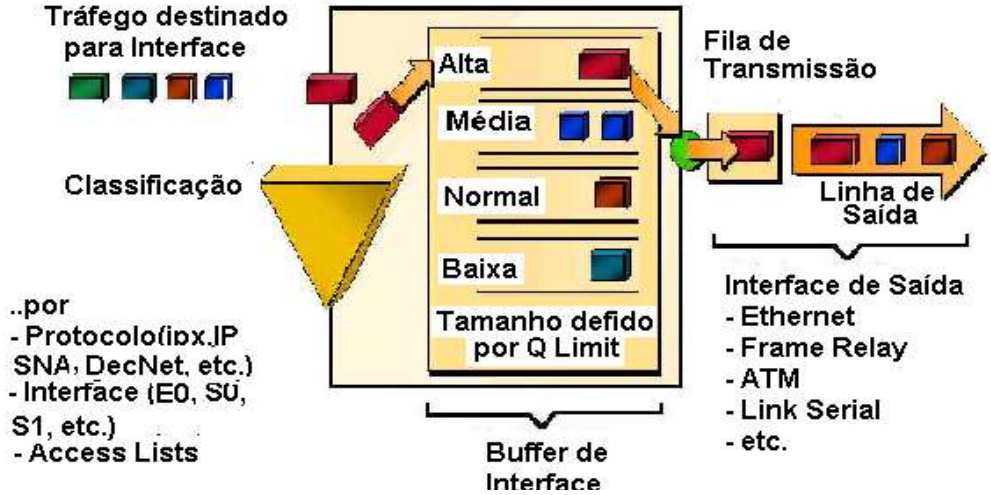


Figura 4-9 - Filas *Priority Queueing*

Numa fila PQ existem várias opções de classificação de tráfego, que podem ser por protocolo (IP, IPX, DecNet, SNA, etc), por interface de entrada ou por lista de acesso.

4.2.6 **Enfileiramento *Custom Queueing(CQ)***

Este mecanismo permite especificar para uma determinada aplicação uma percentagem da banda (alocação absoluta da banda)[7]. O algoritmo da fila CQ (*Custom Queueing*) permite que a banda reservada seja compartilhada proporcionalmente, no percentual pré-definido, entre as aplicações e os usuários. O restante da banda é compartilhado entre os outros tipos de tráfego.

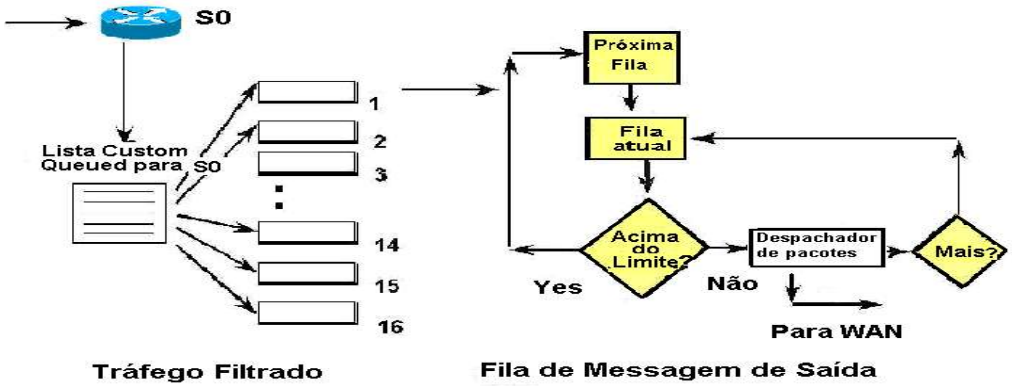


Figura 4-10 - Operação do Enfileiramento *Custom Queueing*

Neste algoritmo o tráfego é controlado pela alocação de uma determinada parte da fila para cada fluxo classificado. Um esquema *round-*

robin é aplicado às filas, que são ordenadas em ciclos, onde uma quantidade de pacotes, referente à parte da banda alocada, é enviado antes de passar para a fila seguinte. Associado a cada fila existe um contador que é configurável. Este contador tem a função de estabelecer a quantidade de *bytes* que devem ser enviados antes da passar para a próxima fila.

Podem ser definidas até 17 filas (figura 4-11), sendo que a fila zero é sempre reservada para mensagens do sistema como sinalização, *keep-alive*, etc. Assim como muitos algoritmos, a classificação CQ pode ser feita através do endereço fonte ou destino, protocolo (IP, IPX, Appletalk, SNA, DecNet, etc), precedência IP, interface de entrada e através de listas de acesso (ACL).

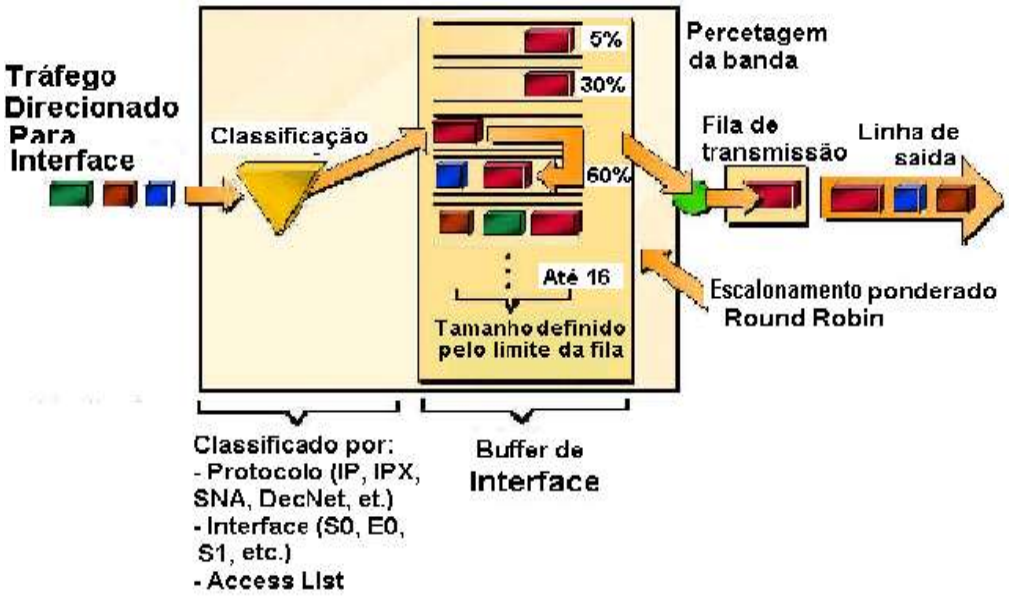


Figura 4-11 - Filas *Custom Queueing*

Abaixo, na tabela 4-2, estão resumidos os três métodos de enfileiramento e a figura 4-12, abaixo, esquematiza algumas considerações sobre como escolher qual método deve ser utilizado. Para um projeto QoS, estas são as diretrizes básicas, mas, como todo projeto tem suas próprias peculiaridades, na prática o que prevalecerá será o método e a configuração dos respectivos parâmetros que se enquadrem nas condições do projeto (disponibilidade de banda nas conexões WAN, topologia do backbone, roteamento estático ou

dinâmico, etc.) e que produza uma boa qualidade subjetiva do sinal.

Weighted Fair Queueing	***Priority Queueing***	***Custom Queueing***
Sem limites de filas	4 filas	16 filas
Prioridade para pequenos volumes	Prioridade alta servida primeiro	Serviço Round-robin
Session dispatching	Packet dispatching	Threshold dispatching
Prioridade para tráfego interativo	Indicado para Tráfego crítico	Alocação da banda disponível
File transfer consegue acesso balanceado	Designado para enlaces de banda estreita	Mais adequado para enlaces de banda larga
Habilitado por default		

Tabela 4-2 - Métodos de Enfileiramento

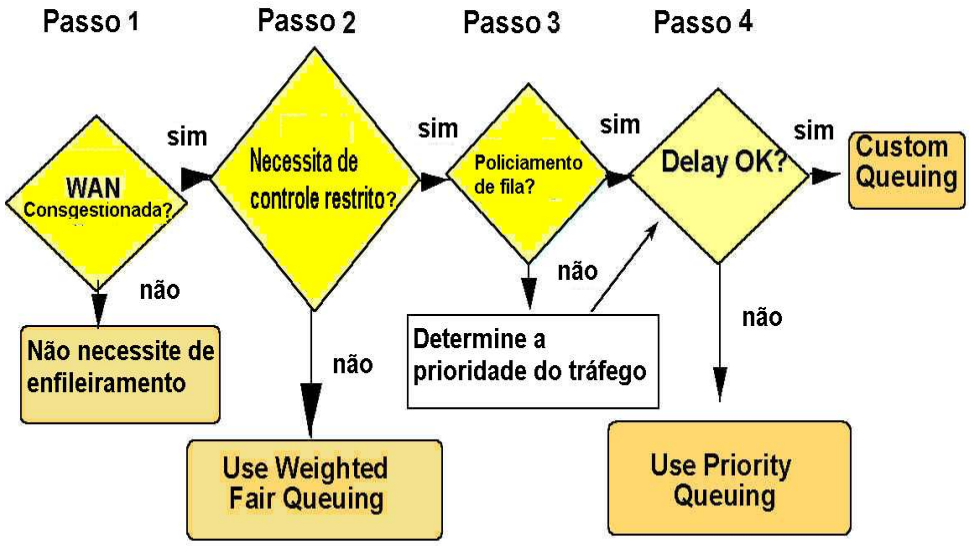


Figura 4-12 - Esquema para Seleção de Método de Enfileiramento

4.2.7 **A Detecção RED**

Para prevenção e inibição de congestionamento foi desenvolvido o mecanismo de detecção RED - *Random Early Detection* (detecção randômica antecipada). Este algoritmo se utiliza das funções de controle de congestionamento TCP e monitora o tráfego antecipadamente, descartando pacotes aleatoriamente e indicando para a fonte reduzir a taxa de transmissão. Com este mecanismo as situações de congestionamento são amenizadas antes que ocorram picos de tráfego. O RED ao ser aplicado a uma interface, descarta pacotes a uma taxa configurável.

Uma variação do RED foi desenvolvida e implementada pela Cisco, o *Weighted RED (WRED)*, esta implementação combina as funcionalidades do RED com a classificação de pacotes por precedência IP. O WRED descarta pacotes de maneira seletiva baseado na classificação por precedência IP, descartando inicialmente os pacotes de menor prioridade, com diferentes pesos para cada classe.

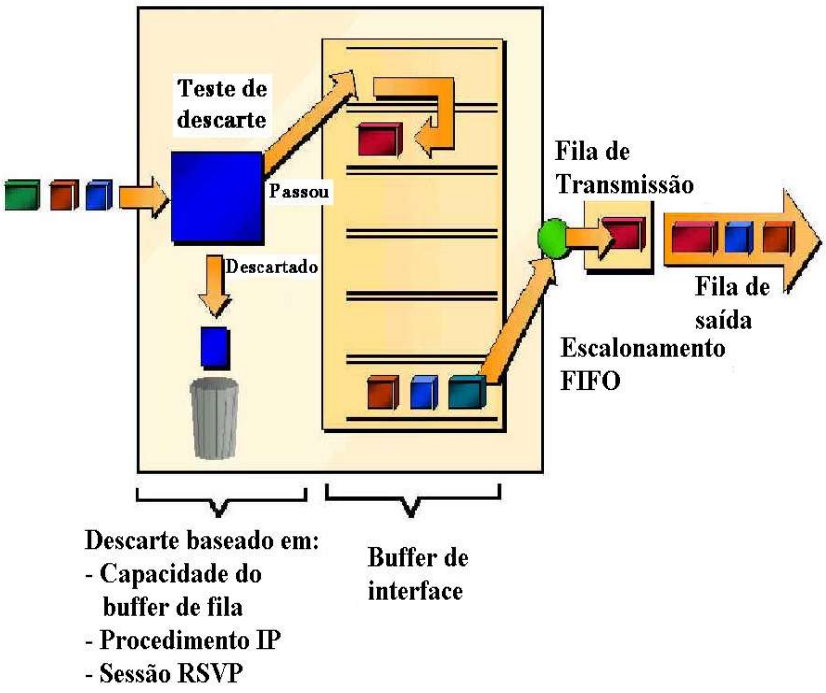


Figura 4-13 - O Funcionamento do WRED

Uma maneira de habilitar o descarte é desabilitar a classificação por precedência IP e habilitar o descarte com base apenas no tamanho do *buffer* da fila; outra maneira é utilizar o WRED em conjunto com o RSVP obtendo um descarte mais seletivo ainda. Assim, os fluxos de menor prioridade nas sessões RSVP serão descartados antes dos outros de maior prioridade, evitando que ocorra uma situação de congestionamento,

O WRED é geralmente utilizado em roteadores centrais de *backbone* (*core routers*), com a precedência IP habilitada pelos roteadores de borda (*edge routers*), mas é um mecanismo útil em qualquer interface onde exista a possibilidade de congestionamento.

4.3 **Moldagem de Tráfego**

Os mecanismos *Generic Traffic Shaping (*GTS), *Class-Based Shaping* (CBS), e *Frame Relay Traffic Shaping* (FRTS) são bastante usados para moldagem de tráfego. Todo os três métodos são similares na execução, embora suas configurações sejam diferentes, e usem tipos diferentes de filas

para conter e dar forma ao tráfego que é encaminhado. O código que determina se um pacote deve ser encaminhado ou se deve ser atrasado é uma característica comum a todos mecanismos GTS. Se um pacote for atrasado, GTS e *Class-Based* usam WFQ para manipular o tráfego, enquanto FRTS usa uma fila *Custom Queue* ou *Priority Queue* para o mesmo fim, dependendo da configuração aplicada. O CBS é um GTS com enfileiramento CBWFQ. Abaixo está descrito o mecanismo GTS, ilustrando o *mecanismo de moldagem de tráfego.*

*O Generic Traffic Shaping (*GTS) [64], provê mecanismos para controle de tráfego utilizando filtros conhecidos como *token bucket (balde de fichas), descrito abaixo*, limitando o tráfego de saída de uma interface a uma determinada taxa. A Figura 4-14 mostra que o tráfego classificado vai para um *buffer* limitador, sendo liberado sob regras pré-definidas de acordo uma política de controle de tráfego, que pode ser configurada pelo administrador ou derivada da interface.

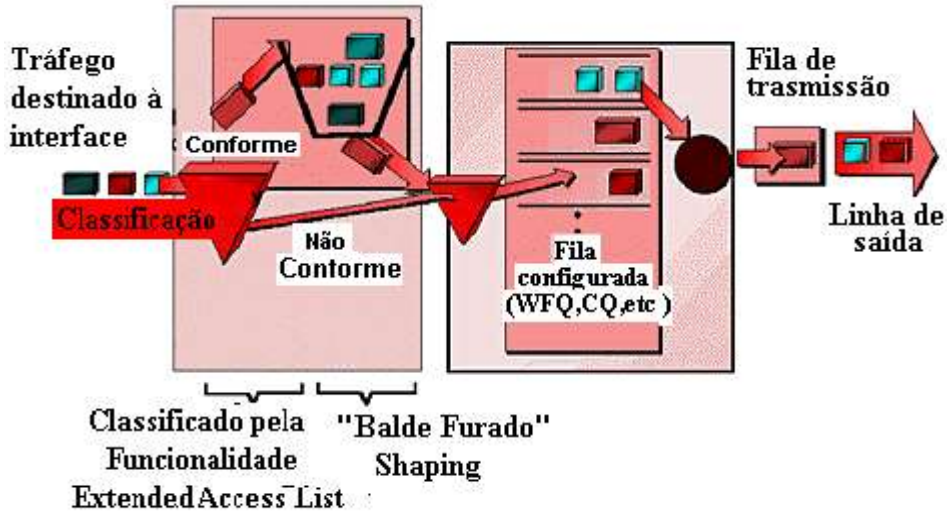


Figura 4-14 - Generic Traffic Shaping

O mecanismo de moldagem ou conformidade de tráfego pode ser utilizado para vários fins. Podemos, por exemplo, proteger o tráfego prioritário limitando o tráfego de rajada, reduzindo assim a latência; ou, em situações de

congestionamento, eliminar possíveis gargalos, limitando um determinado tipo de tráfego não sensível a retardo, como transferência de arquivos.

O mecanismo GTS é aplicado apenas em interfaces de saída, usando listas de acesso para classificar e selecionar o tráfego. Funciona com qualquer tecnologia de enlace (nível 2) como Frame Relay, ATM (*Asyncronous Transfer Mode*), SMDS (*Switched Multimegabit Data Service*) e Ethernet.

4.4 Policiamento do Tráfego

Policiamento compreende o conjunto de regras aplicadas nos nós de borda ao tráfego que entra ou sai do domínio para fazer a verificação de conformidade (*in* ou *out-of-profile*) com os parâmetros de serviço especificados. Todos os fluxos que tenham um determinado par origem/destino são tratados como um fluxo agregado tendo cada um deles um tipo específico de policiamento, medição e DSCP associado. Dependendo do tipo de policiamento em uso e da sua parametrização, o tráfego *out-of-profile* pode ser atrasado, descartado ou remarcado com menor prioridade. Os parâmetros comuns utilizados para a configuração dos mecanismos de policiamento são: *Committed Information Rate (CIR)* e *Peak Information Rate (PIR)* em bits por segundo, *Committed Burst Size (CBS), Excess Burst Size (EBS)* e *Peak Burst Size (PBS)* valores estes em *bytes.*

Existem dois algoritmos bastante utilizados para policiamento de tráfego em redes de pacotes conforme descritos em [47] e [48]. Estes algoritmos são: o método do balde furado (*leaky bucket*) e o método do balde de fichas (*token bucket*). Um outro método comum para policiamento e controle de tráfego IP é o CAR (*Committed Access Rate*) desenvolvido pela Cisco. Abaixo trataremos os dois algoritmos e o CAR.

4.4.1 Método do balde furado

Este método é análogo a um balde com um pequeno furo no fundo, conforme Figura 4-15. Independente da velocidade com que a água entra no balde a sua vazão de saída permanece constante. Quando o balde estiver cheio, a água que entrar vai transbordar e será perdida. Com este mesmo princípio e

usando pacotes ao invés de água Figura 4-16, e estando cada *host* conectado à rede por uma interface que contém um balde furado. Se um pacote chegar e a fila estiver cheia, ele será descartado. O *host* pode inserir na rede pacotes a uma taxa constante, tornando um fluxo de pacotes irregular dos processos de usuário dentro do *host* em um fluxo de pacotes regular para a rede, suavizando rajadas e reduzindo muito as chances de ocorrência de um congestionamento. Se os pacotes forem todos do mesmo tamanho (por exemplo,células ATM), pode-se empregar esse algoritmo como descrito acima. Se os pacotes forem de tamanho variável, o mais indicado é utilizar um número fixo de *bytes* por pulso, em vez de apenas um pacote. Os parâmetros de tamanho do balde e a taxa de transmissão são geralmente configuráveis pelo usuário.

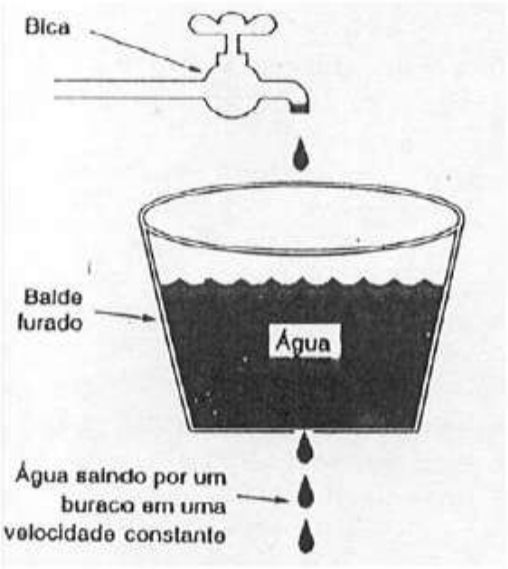


Figura 4-15 Método balde furado (a)

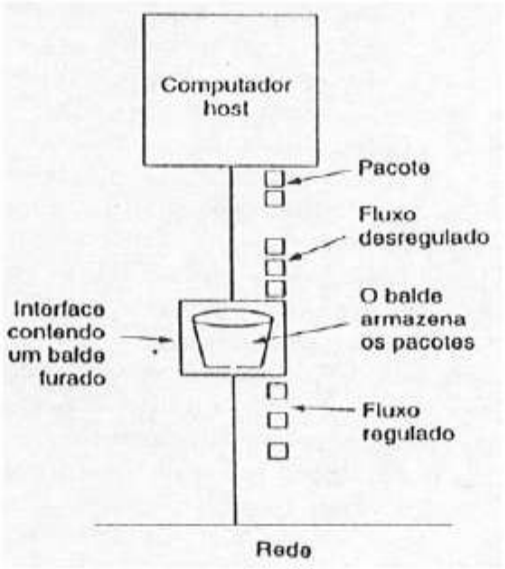
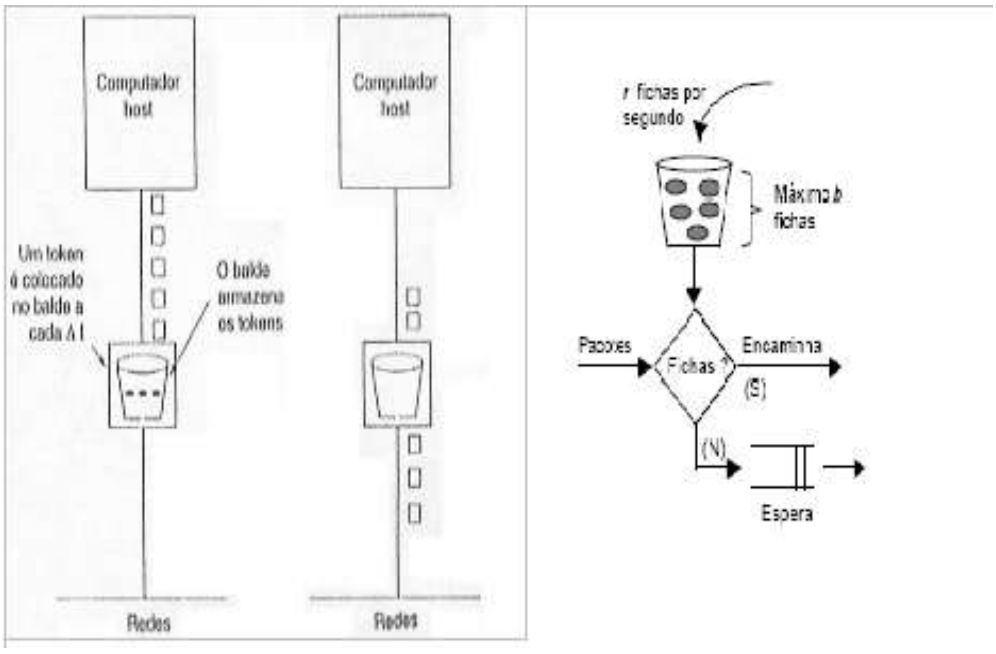


Figura 4-16 Método balde furado (b)

4.4.2 Método do Balde de Fichas

O método do balde furado provoca uma saída constante, independentemente das características do tráfego. Este comportamento muitas vezes não é adequado para algumas aplicações, sendo mais adequado permitir que a saída varie um pouco sua velocidade com o comportamento do tráfego em rajadas. Para suprir essas necessidades existe o algoritmo do balde de

fichas, onde o balde retém fichas, gerados por um *clock* na faixa de uma ficha cada variação de T segundos. Nesta analogia para que um pacote seja transmitido, ele deve capturar e destruir um ficha. Na Figura 4-17(a) é mostrado um balde com três fichas e cinco pacotes aguardando para serem transmitidos. Na Figura 4-17(b), mostra um momento seguinte em que três dos cinco pacotes percorreram todo o caminho, mas dois ficaram retidos aguardando que ficha seja gerada. Quando o balde enche, este algoritmo descarta as fichas geradas, mas nunca descarta pacotes. Uma idéia para tornar o tráfego mais regular seria colocar um balde furado após o balde de fichas. Devido à característica do balde de fichas permitir rajadas em pequenos intervalos de tempo.



(a) (b) (c)

Figura 4-17 - Método balde de Fichas

O token bucket (balde de fichas) possui três parâmetros: um tamanho de rajada em bits (*burst size*), também chamado de *committed burst* (Bc) que especifica o quanto pode ser enviado num determinado intervalo de tempo; uma taxa média (*mean rate*) em bps, também chamada de CIR (*Committed*

Information Rate), que especifica o quanto de dados, em média, pode ser enviado por unidade de tempo; e um intervalo de tempo (Tc) em segundos, intervalo de medida que especifica o tempo por rajada. A analogia é imaginar um balde com uma determinada capacidade máxima que contém fichas que são inseridas regularmente. Uma ficha corresponde à permissão para transmitir uma quantidade de bits. Quando chega um pacote, o seu tamanho é comparado com a quantidade de fichas no balde. Se existir uma quantidade suficiente de fichas, o pacote é enviado. Senão, geralmente é inserido em uma fila para que aguarde até haver fichas suficientes no balde. Isso é chamado de suavização ou moldagem de tráfego. Caso o tamanho da fila seja zero, todos os pacotes fora do perfil (que não encontram fichas suficientes) são descartados.

4.4.3 **CAR *Committed Access Rate***

Um método comum para policiamento e controle de tráfego IP é o CAR (*Committed Access Rate*) desenvolvido pela Cisco. O CAR tem como objetivo oferecer ao operador da rede a facilidade de gerenciar a alocação de largura de banda que foi determinada na criação da conexão. O tratamento ocorre através de *thresholds* e pode ocorrer tanto em filas de entrada como em filas de saída dos roteadores. Este método realiza, basicamente, duas funções para qualidade de serviço:

- O *policing,* que é o gerenciamento de banda com limitação de taxa de acesso - este gerenciamento permite controlar a taxa máxima de transmissão ou recepção de dados de uma determinada interface. O mecanismo permite que o tráfego em conformidade seja transmitido ou recebido, enquanto que os pacotes que excedem os limites pré-definidos, podem ser descartados, ou podem ser reclassificados com outra prioridade para retransmissão;
- Classificação – esta classificação de pacotes pode ser através de precedência IP ou de grupos de QoS (um rótulo interno do roteador utilizado para definição de classes) e permite particionar a rede em

múltiplas classes de serviços ou níveis de priorização.

O CAR utiliza o mecanismo *token bucket* para examinar o tráfego recebido na interface ou parte do tráfego selecionado pelos critérios das listas de acesso, compara a taxa de tráfego com a do *token bucket* e, de acordo com o resultado, toma uma ou várias ações. O CAR pode transmitir, descartar ou reclassificar o pacote com outro nível de prioridade.

A figura 4-18 mostra graficamente a ação do CAR. Os *tokens* são inseridos no balde na mesma taxa CIR. A profundidade do balde é o tamanho da rajada (*burst size*). Se houver *tokens* suficientes quando o tráfego chega ao balde, então o tráfego é dito estar em conformidade e a quantidade correspondente de *tokens* é removida. Se não houver *tokens* suficientes, então o tráfego é dito excessivo.

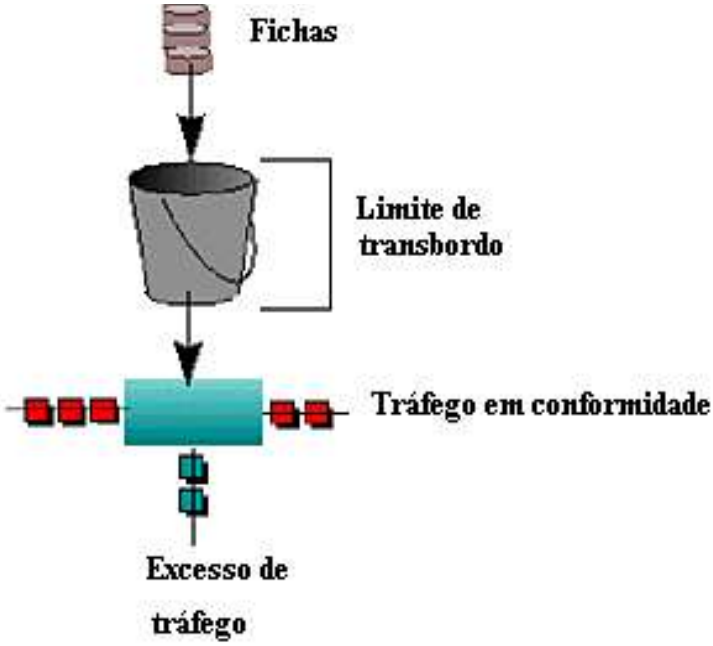


Figura 4-18 - *Commited Access Rate* com *token bucket*

Como exemplo de CAR *Policy* podemos citar[49]:

- *Firm* CAR – pacotes que excedem a banda alocada são descartados.
- CAR + *Premium* – pacotes que excedem a banda alocada são

reclassificados com alta ou baixa preferência.

- CAR + *Best Effort* – pacotes que excedem a banda alocada são reclassificados até o estouro do *threshold*, depois são descartados.
- *Per Application* CAR – diferentes CARs são especificados para diferentes aplicações.

Os critérios de seleção de tráfego podem ser baseados em todo tráfego IP, em precedência IP, em listas de acesso (padrão ou estendida), em endereço MAC ou, ainda, em grupos de QoS. O endereço MAC e a precedência IP podem ser definidos através de listas de acesso *rate-limit*.

Com o CAR, é possível limitar o tráfego por aplicação (Web, FTP, etc), por interface (p. ex., uma conexão serial a 2 Mbps, mas com acesso limitado em 512 Kbps), por endereço MAC (p. ex., controle de tráfego em PTT - Pontos de Troca de Tráfego), ou pode-se classificar ou reclassificar todo o tráfego de entrada num *backbone* a partir dos roteadores de borda, para tratamento diferenciado em termos de qualidade de serviços.

CAR e GTS utilizam o mecanismo *token bucket* (balde de fichas) para realizarem suas funções, mas guardam características e aplicabilidades distintas. Enquanto GTS é usado para moldagem de tráfego o CAR é utilizado para policiamento e classificação de tráfego. Abaixo, na Tabela 4-3, é mostrado as principais diferenças entre estes mecanismos.

CAR	GTS
Aplica-se em tráfego de entrada e de saída	Aplica-se apenas em tráfego de saída
Não "buferiza" nem molda tráfego	Molda e suaviza o tráfego
Pode marcar pacotes (ex: precedência IP)	Não possui essa funcionalidade
Em Frame Relay, não suporta FECN e nem BECN	Suporta FECN e BECN em Frame Relay
Suporta políticas em cascata	Não suporta essa funcionalidade
Não suporta uso com RSVP	Suporta RSVP quando usa WFQ
Suportas listas padrões e estendidas	Suportas apenas listas estendidas
Gerência o descarte entre o *normal_burst* e o *extended_burst*	Não suporta essa funcionalidade
Executa no modo distribuído (VIP do 7500)	Não suporta essa funcionalidade

Tabela 4-3 - Comparação entre CAR e GTS

4.5 Controle de Admissão

O Controle de admissão implementa o algoritmo que um roteador usa para determinar se um novo fluxo pode ter seu pedido de **QoS** atendido sem interferir nas garantias feitas anteriormente. É algo semelhante ao que ocorre no sistema telefônico, onde nós ouvimos um “sinal de ocupado” quando o sistema não tem recursos disponíveis para atender a chamada que está sendo feita. Nesse caso, significa que alguns fluxos podem ter seus pedidos de recursos rejeitados por falta de recursos em algum dos roteadores.

O controle de admissão é um mecanismo essencial para o *IntServ*, uma vez que *o* RSVP, para efetuar a reserva de recursos num roteador, se comunica com os módulos locais de controle de admissão e este mecanismo determina

se o nó tem capacidade para fornecer a QoS requisitada.

4.6 Controle de Congestionamento

Congestionamentos em redes de computadores são basicamente um problema de compartilhamento de recursos [46]. Estes congestionamentos ocorrem quando são inseridos na rede uma quantidade maior de pacotes do que ela é capaz de tratar. Para tratar congestionamentos, existem duas atividades distintas relacionadas ao controle ou gerenciamento de congestionamento [43], [44], [41]. A primeira é a prevenção de congestionamento que tenta detectar possíveis condições que levem a congestionamentos futuros e executar procedimentos para impedi-los. A segunda é a recuperação de congestionamento, que atua na rede quando um congestionamento já ocorreu para que ela volte ao seu estado normal. Hoje, na Internet o controle de congestionamento é feito somente pelo protocolo TCP na camada de transporte. A taxa de transmissão de dados do protocolo TCP é controlada pelas condições da rede. Os fluxos TCP respondem positivamente às notificações de congestionamentos, por isso são chamados de "compreensivos" ou "responsáveis". O protocolo UDP, ao contrário, não possui nenhum mecanismo para controlar congestionamentos. Assim, o UDP não altera sua taxa de transmissão, só as aplicações podem interferir no comportamento do fluxo UDP[40]. Por este motivo, os fluxos UDP são chamados de "agressivos" ou "não compreensivos".

Segundo [45], cerca de 90% dos pacotes na Internet pertencem a fluxos TCP, em geral os congestionamentos são resolvidos de maneira adequada, apesar de várias implementações TCP existentes estarem incorretas [39].

O mecanismo de controle de congestionamento monitora o fluxo de cargas da rede em um esforço de antecipar e evitar que congestionamentos comuns da rede interna se transformem em problema. Estas técnicas são projetadas para fornecer o tratamento preferencial para o tráfego alta prioridade sob situações de congestionamento, isto acontece ao simultaneamente maximizar o *throughput* da rede e minimizar a perda de

pacotes e *delay*. A família RED oferece mecanismos característicos para controlar congestionamento.

Os algoritmos de controle de congestionamento do protocolo TCP (que foram criados por Van Jacobson [41]), são padronizados pela RFC 1122 e são apresentados com mais detalhes na RFC 2581 [38]. São quatro os algoritmos utilizados pelo TCP: *slow start*, *congestion avoidance*, *fast retransmit* e *fast recovery*. O objetivo global deles é levar a rede a um estado estável de plena utilização, permitindo a introdução de um novo pacote à medida que outro pacote é retirado. Desse modo, os recursos na rede não são desperdiçados, ao mesmo tempo em que os congestionamentos são evitados. Cada transmissor TCP fica o tempo todo monitorando a rede, tentando transmitir o máximo de segmentos possível.

Estes capítulos iniciais deram uma visão generalizada de QoS em redes IP, descreveram os requisitos essenciais e os mecanismos mais conhecidos para implementação de QoS. Os próximos capítulos tratarão de QoS voltado para um ambiente específico de uma determinada rede, descreverá as necessidades, os mecanismos, as particularidades e os experimentos em laboratório. Também serão mostrados os resultados obtidos e as conclusões tiradas.

4.7 Considerações sobre Este Capítulo

Este capítulo abordou os mecanismos necessários para o provimento de QoS em uma rede. Foram tratadas cada uma das fases que o fluxo de dados é submetido para o provimento de QoS e os mecanismos empregados em cada uma dessas fases, além das descrições dos algoritmos envolvidos em cada mecanismo. O próximo capítulo tratará das tarefas envolvidas na provisão de QoS em domínio *DiffServ*.

5 PROVISÃO DE QOS EM UM DOMÍNIO *DIFFSERV* (DS)

Devido às características do ambiente alvo deste trabalho, onde as aplicações prioritárias tem comportamentos conhecidos e necessidades de recursos moderadas, onde não há necessita de grande investimento em hardware e software, pois já estão presentes nos roteadores da Previdência, dentre as alternativas técnicas disponíveis a que reuniu mais benefícios e menos impactos foi a alternativa *DiffServ*. Esta solução também apresenta uma alta escalabilidade e um moderado impacto tecnológico. Este trabalho foi desenvolvido empregando esta tecnologia utilizando, para sua implementação, roteadores Cisco e mecanismos de QoS presentes em seu sistema operacional (IOS Cisco). Este capítulo descreve um domínio DiffServ, trata das tarefas necessárias à provisão de serviços em um domínio DiffServ (DS), descreve as funções dos principais componentes de um DS e faz observações sobre os mecanismos disponibilizados pela Cisco.

5.1 O domínio DiffServ.

Um Domínio DiffServ é composto por um grupo de nós contíguos que operam um conjunto comum de serviços[67]. O princípio desta arquitetura é trazer as operações mais complexas para as bordas do domínio, aumento a sua escalabilidade, deixando o interior do domínio com mecanismos mais simples. Para o domínio *DiffServ* é válido afirmar que há um estado por fluxo nos nós de borda do domínio e um estado por classe de serviço nos nós internos ao domínio.

A Figura 5.1 ilustra um domínio *DiffServ*. Nas extremidades do domínio estão localizados os nós de fronteira 1, 4 e 5. Nesses nós há uma agregação de fluxos que são encaminhadas através dos nós interiores 2 e 3. Nesta arquitetura a diferenciação de serviços é provida de maneira assimétrica, isto é, em uma única direção do fluxo de tráfego. Conforme o sentido do fluxo, os nós de fronteira assumem o papel de nós de ingresso ou egresso do domínio conforme mostrado na figura 5.1, onde o nó 1 representa um roteador de

ingresso e o nó 4 um roteador de egresso.

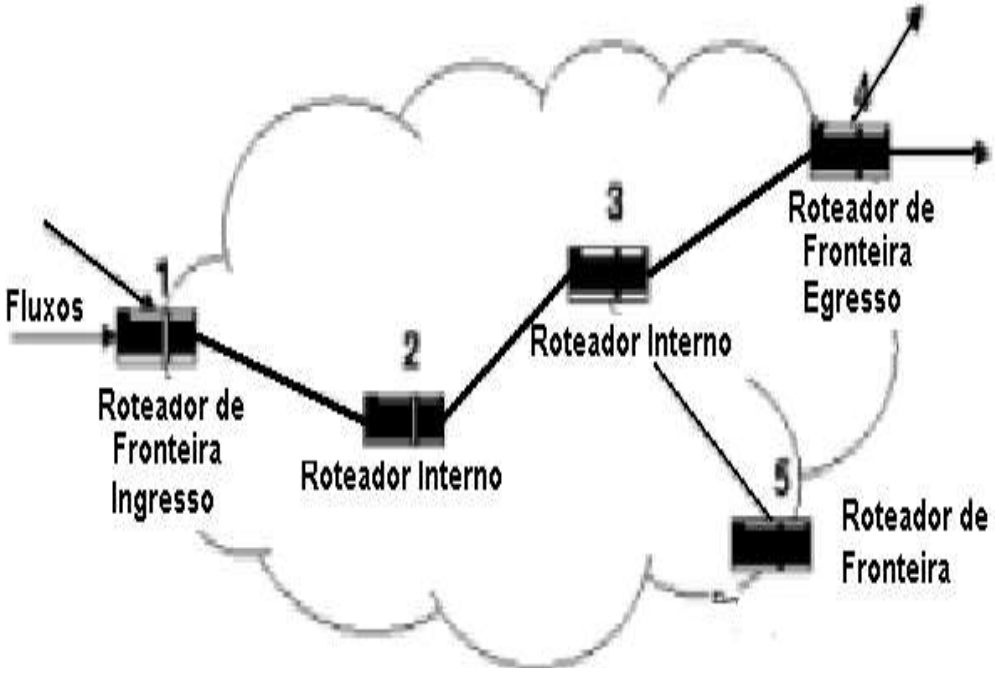


Figura 5.1. Domínio de Serviços Diferenciados – DS

A implementação de QoS em um DS necessita que algumas tarefas sejam executadas. Abaixo estão descritas as tarefas referentes à provisão do serviço em um Domínio DiffServ (DS).

5.1.1 Controle de Admissão

O controle de admissão é responsável pelo controle do fluxo na entrada do DS. Este controle é feito através de ACLs no roteador de borda mais próximo da fonte de tráfego utilizando, por exemplo, endereços IP fonte e destino ou endereço da porta fonte/destino. Caso o pacote não passe pelo controle de admissão é remarcado para o serviço de melhor esforço.

5.1.2 Classificação

A classificação provê um serviço preferencial a um determinado tipo de tráfego. Consiste na seleção de pacotes baseada, ou em vários campos contidos num pacote IP (*Multi-Field Classification – M*F), como endereços origem e destino, ou no campo DSCP (*DS CodePoint*). Enquanto a classificação MF é

habitualmente aplicada a fluxos individuais, a classificação DSCP é aplicada a tráfego agregado previamente marcado (*Behaviour Aggregate Classification - BA*). Classificação é a base para a aplicação das políticas que selecionam pacotes que atravessam um simples elemento de rede ou passam por uma particular interface envolvendo diferentes tipos de serviços QoS. O Cisco IOS provê as ferramentas [42]: *Policy-Based Rounting (PBR),* que oferece uma maneira flexível de rotear pacotes por permitir a configuração de políticas para um determinado fluxo; *QoS Policy Propagation via Border Gateway Protocol (BGP),* que utiliza o protocolo BGP para fazer classificação; *Cmmitted Access Rate (CAR)*, que implementa classificação através de taxas limites; e *Class-Based Packet Marking*, que é uma ferramenta para marcação de pacotes baseada em *IP precedence*, DSCP, CoS, *QoS Group e Cell Loss Priority* (CLP) atualizando o bit ATM do cabeçalho do pacote. A classificação do tráfego é feita no roteador de borda mais próxima da fonte. Estes mecanismos suportam as RFCs 2474 [22], 2475 [2], 2597 [36] e 2598 [42].

5.1.3 Medição

Esta é uma funcionalidade que mede as propriedades temporais do tráfego previamente classificado. Os dados desta medição são confrontados com os perfis de tráfego especificados no *Traffic Conditioning Agreement (TCA)*. É, também, utilizada por outras funções de TC para regular o tráfego conforme este esteja *in-profile* ou *out-of-profile*.

5.1.4 Marcação

A marcação é a base da arquitetura DS porque identifica os pacotes possibilitando a sua classificação. É realizada através da marcação do campo IP *DS-field* com um valor específico (DSCP) [50]. As regras do TCA baseiam a marcação/remarcação dos pacotes IP de modo a especificar os PHBs que devem ser aplicados. A marcação de pacotes é feita atribuindo valores aos DSCPs com técnicas baseadas em classes.

5.1.5 Policiamento

O policiamento dos fluxos é feito no roteador de borda mais próximo da fonte de tráfego através do DSCP e da capacidade de agregação do link contratado para o par de instituições fonte/destino. O policiamento verifica a conformidade do tráfego com um perfil específico, elimina ou remarca o tráfego não conforme para prioridades mais baixas. Por exemplo, enquanto os pacotes AF sem conformidade com as características especificadas para ele (*out-of-profile)* podem ser remarcados para aumentar a sua probabilidade de descarte, os pacotes EF excedentes podem ser imediatamente eliminados. Os mecanismos Cisco IOS para policiamento de tráfego são [51]: CAR, que utiliza taxas limites para policiamento de tráfego, e o *Traffic Policing*, que permite controlar a taxa máxima de tráfego enviado ou recebido em uma interface utilizando o algoritmo *Token Bucket* (TB).

5.1.6 Shaping

O *Shaping* ajusta o tráfego para um perfil definido, esta funcionalidade regula o tráfego que é submetido a um domínio DS. O *Shaping* pode atrasar os pacotes de forma a ajustar as suas características ao perfil definido. O *Token Bucket* (TB) e o *Leaky Bucket* (LB) são os algoritmos mais utilizados para implementar *Shaping*. Cisco IOS QoS possui três tipos de *traffic* shaping [51]: *Generic Traffic Shaping* (GTS), que evita congestionamento pondo o tráfego dentro de uma taxa de transferência usando o mecanismo TB; *Class-Based Shaping*, que faz o policiamento baseado em classe de trafego de acordo com as políticas configuradas e *Frame Relay Traffic Shaping*, que utiliza parâmetros do *frame relay* para fazer o controle de tráfego.

5.1.7 Escalonamento

O escalonamento é o mecanismo responsável pelo comportamento por salto (PHB) em cada roteador do domínio DS. Os mecanismos que melhor implementam o serviço, são o PQ (*Priority Queuing*) que possibilita a prioridade absoluta para um determinado tráfego e o WFQ (*Weighted Fair Queue*) que trata com maior justiça os serviços.

5.1.8 Controle de Congestionamento

O objetivo do controle de congestionamento é o de prevenir que este aconteça. A maneira mais utilizada para este controle é o algoritmo RED (*Random Early Detection*), proposto por *Jacobson e Floyd* [58], que é uma referência neste objetivo. Este mecanismo elimina pacotes excedentes de acordo com um determinado critério de prioridade. Existem várias variantes do RED como o WRED, o GRED, o RIO-C e o RIO-DC, apresentados em [59]. As extensões do RED com suporte multi-nível (vários *Drop Precedence (DP)*) são adequadas à implementação dos DPs das classes do PHB AF. O mecanismo *Tail Drop* também é usado, este mecanismo trata o tráfego de maneira igual em todas as classes e elimina pacotes sem critério de prioridade.

5.1.9 Segurança

A principal medida de segurança é realizada no controle de admissão ao serviço onde só são permitidos acessos por regra explícita na lista de controle de acesso (ACL) do roteador.

5.2 Funcionalidades *DiffServ*

As funcionalidades *DiffServ* compreendem as funções dos componentes de um DS e as tarefas executadas por estes para realização de suas funções. Estas funcionalidades estão distribuídas nos componentes de borda e de centro conforme descritos abaixo.

5.2.1 Componente de Borda

As funcionalidades dos componentes de borda são: Classificação, Marcação, Medição, Policiamento e *Shaping*. Estas funcionalidades podem ser aplicadas a fluxos individuais de tráfego, geralmente nos nós de ingresso (*ingress nodes*) ou a tráfego agregado, geralmente nos nós de ingresso/egresso

(*ingress e egress nodes*). Embora estas funcionalidades sejam conceitualmente distintas, desde o ponto de vista de uma implementação, podem estar agrupadas [53].

5.2.2 Componente de Centro

O componente de centro recebe o fluxo já marcado pelos componentes de borda, por isso é mais simples e requerendo apenas as funcionalidades de Classificação BA e *Queuing*. Este é o aspecto que mais favorece a escalabilidade da arquitetura *DiffServ*.

- **Classificação *Behaviour-Aggregate(BA)*** – Esta classificação utiliza apenas o campo IP *DS-Field* com tráfego já agregado, o que a torna mais simples do que a classificação MF. Isto simplifica o processamento no nó de centro.
- ***Queuing*** – O *queuing* executa tarefas de armazenamento, descarte e escalonamento de pacotes. Os PHBs necessitam, para sua implementação, que existam mecanismos de alocação e gestão de *buffers*, e mecanismos de escalonamento. Para que sejam implementadas as classes de serviços(CoS), geralmente o tráfego é dividido em filas de espera separadas e tratado por disciplinas que selecionam os pacotes a serem enviados de acordo com a QoS pretendida. O *Priority Queuing* (PQ) [52] e o *Fair Queuing* (FQ) [52], são exemplos de abordagens possíveis na implementação de disciplinas de *Queuing* para fornecer QoS, além do FIFO, que por si só não permite a diferenciação do tráfego. A PQ implementa prioridade estrita entre as filas. Isto significa que as filas de baixa prioridade só serão servidas quando as filas de maior prioridade não tiverem tráfego para enviar. Este mecanismo pode causar problemas de postergação indefinida e/ou *burstiness* ao longo da rota [12], [56], embora seja muito interessante para implementar o PHB EF. O FQ controla a largura de banda atribuída a cada classe

tentando minimizar as limitações do PQ. Existem algumas variantes do FQ, como o CBWFQ, que está especialmente orientado para lidar com diferentes CoS [57].

5.2.3 Particularidades da Implementação Cisco

A plataforma Cisco apresenta uma maneira bastante simples de programação, um número significativo de funcionalidades *DiffServ*, mecanismo de TC (Marcação, *Policing* e *Shaping*), uma boa documentação e é bastante utilizada no mundo.

Os métodos utilizados para identificação de fluxos pela implementação Cisco incluem *access control list (ACL), policy-based rounting (PBR), commited access rate (CAR), e network-based aplication recognition (NBAR).*

A configuração da Classificação *Multi-Field (MF)* e Marcação são feitas juntas. A implementação Cisco utiliza o *Policy-Based Routing* (PBR), que permite a classificação de pacotes baseada em múltiplos campos.

O Policiamento utiliza o CAR (*Committed Access Rate*) [57], podendo o tráfego excedente ser eliminado ou remarcado.

Shaping é a funcionalidade implementada pela Cisco através do *Generic Traffic Shaping* (GTS). Este mecanismo segue o algoritmo TB (*Token Bucket*).

A funcionalidade Classificação *Behaviour-Aggregate (BA***)** encontra-se embutida no *Queuing.* O *Queuing* inclui três algoritmos diferentes, além do FIFO. Um é o PQ, que implementa um esquema de prioridade estrita entre as filas. Isto significa que as filas de baixa prioridade só serão servidas quando as filas de maior prioridade não tiverem tráfego para enviar. Este mecanismo pode causar problemas de *starvation* e/ou *burstiness*. A maior limitação deste algoritmo é que apenas quatro classes podem ser definidas. O segundo algoritmo é o CQ (*Custom Queuing*), onde uma percentagem da largura de banda total é alocada à classe. Finalmente o WFQ (*Weighted Fair Queuing),* que controla a largura de banda atribuída a cada classe tentando minimizar as limitações do PQ. O WFQ divide-se em duas variantes: o FB-WFQ (*Flow-*

Based Weighted Fair Queuing), que implementa o escalonamento dinâmico utilizando heurísticas próprias (sem necessidade de configuração), não sendo adequado à implementação de *DiffServ,* e o CBWFQ (*Class-Based Weighted Fair Queuing*) [57], que permite a implementação de todos os PHBs que foram definidos até agora. No CBWFQ é possível garantir para cada classe um mínimo de largura de banda (útil para implementar classes AF), sendo o resto da largura de banda distribuída proporcionalmente às reservas efetuadas. É ainda possível definir uma classe que tenha prioridade absoluta sobre as restantes. A definição desta classe pode ser crítica já que pode provocar *starvation*. No entanto, com algum cuidado, o CBWFQ é uma boa solução para implementar EF, AFs e BE simultaneamente em Cisco. Além do CBWFQ, outras foram utilizadas para implementar o grupo de PHBs AF e em alguns casos EF e AFs [10, 12].

O Controle de Congestão é um mecanismo que no Cisco é implementado utilizando WRED (*Weighted RED*). Esta implementação baseia-se no algoritmo RED e permite a definição de múltiplos precedências de descartes. O WRED é parametrizado definindo um Min, Max e a correspondente precedência de descarte. no ponto Max.

5.3 Considerações sobre Este Capítulo

No início deste capítulo há uma justificativa para uso do *DiffServ* neste trabalho e a descrição de um domínio *DiffServ*. Em seguida são descritas as tarefas referentes à provisão de QoS em um Domínio *DiffServ*. Essas tarefas determinam a aplicação das políticas estabelecidas para o provimento dos serviços de maneira diferenciada, conforme um projeto criado para implementar uma solução envolvendo QoS. São tarefas que manipulam os fluxos agregando-os em classes e dando-lhes tratamento diferenciado e, assim, otimizando o uso dos recursos de rede. Este capítulo também descreveu a função de cada componente que compõe um Domínio *DiffServ*(DS), envolvendo a aplicação dos mecanismos de QoS que, conforme função, cada componente necessita para implementar as políticas estabelecidas. Descreveu,

também, as particularidades dos mecanismos desenvolvidos pela Cisco e as ferramentas disponíveis para sua implementação. O capítulo seguinte tratará da montagem do ambiente de teste, configurações de equipamentos e aplicação de testes.

6 O AMBIENTE DE TESTE

Os capítulos anteriores mostraram toda parte teórica referente à implantação de QoS utilizando a tecnologia de Serviços Diferenciados (*DiffServ*). Mostraram os conceitos, as tecnologias disponíveis, os requisitos essenciais, os mecanismos, as características de um DS e as facilidades disponíveis de *hardware* e *software* que compõe a rede em estudo. Este capítulo começa com a caracterização do tráfego prioritário de um posto do INSS, onde foram levantados os parâmetros para a montagem da planilha de testes, depois, com base nestes estudos e levantamentos, foi montado um ambiente de teste onde estão retratadas as situações mais próximas da realidade, levando-se em consideração a plataforma existente, a escassez de recursos e a atualização tecnológica necessária(*hardware e software*). Foram utilizados nos laboratórios os mesmos equipamentos que são utilizados na rede da Previdência. Neste capítulo é descrito o ambiente de teste, onde são mostrados a topologia da rede, os componentes e o domínio diferenciado, assim como os métodos utilizados nos experimentos e as ferramentas de software adotadas no testbed.

6.1 Caracterização do Tráfego

Os postos de trabalho do INSS são caracterizados por prestarem serviços de benefícios a segurados e serviços de arrecadação aos contribuintes do INSS utilizando softwares específicos baseados no paradigma cliente/servidor. A carga de trabalho apresentada pelo cliente consiste de requisições de serviços a servidores remotos e suas respectivas respostas. Os servidores estão geograficamente distribuídos na rede e podem prestar serviços a uma região (servidores regionais), a um estado (servidores dos escritórios estaduais) ou a toda corporação (servidores corporativos).

A pretensão deste trabalho não é caracterizar todo o tráfego que entra ou sai de um determinado nó, o objetivo é permitir que determinados fluxos tenham garantia de fluência nos momentos de congestionamentos. Desta

maneira o trabalho de caracterização de tráfego está resumido a identificação das características dos fluxos responsáveis por manterem os serviços essenciais de atendimento a clientes nos postos do INSS. Com esta caracterização os parâmetros para montagem da planilha de testes são identificados.

O tráfego prioritário para os enlaces foi determinado pela necessidade do atendimento de benefícios e de arrecadação no postos remotos de INSS. Estas preferências estão bem relacionadas às características de cada posto e a sua função dentro da prestação de serviços à comunidade. Desta maneira são encontradas algumas variações nos fluxos prioritários concorrentes em cada enlace, tanto na diversidade quanto na intensidade de utilização. Como são conhecidos origens e destinos dos fluxos prioritários, o trabalho de caracterização da carga de trabalho está resumida à observação do comportamento dos fluxos originados e recebidos por determinados *hosts* identificados por seus endereços.

A caracterização dos fluxos foi conseguido através da observação, via softwares monitores, dos fluxos de entrada e de saída dos roteadores de borda. Depois de observados exaustivamente o comportamento dos fluxos, as medidas foram registrados através da utilização de:

- Comandos CLI (*command line interface*) nos roteadores Cisco - Utilizado para quantificar o consumo de cada fluxo;
- NTOP e *Sniffer* da NAI [63] - Utilizado para verificação do tipo de tráfego, na identificação dos fluxos e na geração de estatísticas para esta caracterização do tráfego;
- MRTG – Utilizado para visualização do comportamento do tráfego nos diversos enlaces.

Como QoS é unidirecional e os aplicativos são bidirecionais, houve uma

necessidade de caracterizar o tráfego nos dois sentidos para identificar as necessidades de QoS das requisições e das respostas. Os parâmetros observados para caracterização de cada fluxo que compõem o tráfego foram: Tráfego em ambos os sentidos, tamanho de pacotes de cada fluxo em ambos os sentidos e tipo de protocolo. Os parâmetros de quantidade de fluxos simultâneos, comportamento de cada fluxo com relação às suas características de operação e sua prioridade relativa aos serviços prestados pelo INSS, são parâmetros dependentes de cada posto e serão utilizados no momento de estruturar a implementação dos mesmos. A tabela 6.1 mostra os fluxos prioritários e suas características. É observado que a exigência de banda no sentido borda para o centro é bem menor que a necessidade no sentido inverso. Isto deve ser observado na especificação dos contratos de enlaces junto às prestadoras de serviços e nos cálculos para garantia de QoS.

Aplicação	Protocolo usado	Necessidade de BW/fluxo (kbps)		Tráfego Entrada		Tráfego Saída	
		In	Out	Tam (bytes)	Quant (pcts)	Tam (bytes)	Quant (pcts)
PRISMA (Benefícios)	TELNET	3,79	0,43	380	1496	40	1600
SIPPS (Protocolo)	HTTP	3,01	0,13	1130	400	165	117
PLENUS CV3 (Benefício)	TELNET	0,62	0,13	347	267	68	288
PLENUS MV2 (Arrecadação)	TELNET	0,76	0,15	371	307	80	277
CNIS (Cadastro Nacional)	HTTP	2,38	0,90	700	510	270	500
SABI (Serviço Médico)	TCP	11,22	8,15	204	8253	154	7940
Servidor de Nomes (WINS)	WINS	0,06	0,08	92	94	82	140
Servidor de Nomes (DNS)	DNS	0,08	0,04	127	94	65	98

Tabela 6.1 – Aplicativos essenciais para um posto de atendimento INSS

6.2 Equipamentos e Softwares para o laboratório

Para montagem do laboratório foram utilizados os seguintes recursos de *hardware* e *software*:

- Sistema Operacional das estações - Linux Kernel 2.4.21 – Distribuição Conectiva Linux 9.0;
- Gerador/Medidor de tráfego UDP -- IPERF/MGEN/DREC/MCALC;
- Gerador/Medidor de tráfego TCP – Netperf/Netserver;
- Visualização de pacotes – Netpeek;
- Monitoramento de Tráfego – MRTG;
- 03 roteadores Cisco modelo 2600 com Sistema Operacional IOS Cisco 12.1(5)T;
- 03 computadores para geração e recebimento de tráfego;
- 02 cabos DTE/DCE Cisco;
- 02 cabos de par trançado *Cross-Over* (01 de redundância);
- 04 cabos de par trançado (redundância de 02);
- 01 Hub de oito portas.

Algumas premissas foram obedecidas para que o experimento oferecesse um ambiente estável e medidas confiáveis. Estas premissas estabelecidas para o bom funcionamento do experimento foram:

- Alocação de um mínimo de 25% da banda para a classe *Best Effort*;

- Marcação dos pacotes que entram na rede para evitar tráfego espúrio;
- Classificação do tráfego no núcleo da rede somente pelo DSCP;
- Obediência às recomendações da Cisco para configuração das classes de tráfego.

6.3 Descrição do Testbed

A plataforma de testes é composta por três microcomputadores *Pentium*, nos quais foram instalados Sistemas Operacionais (SO) Linux[1], três roteadores Cisco[2] com suporte a QoS conectados por cabos DCE/DTE Cisco.

A figura 6-5 mostra a topologia adotada e o domínio *DiffServ* (DS). A estrutura é montada numa rede *Ethernet* (IEEE 802.3 10BaseT/100BaseT), os roteadores conectados por cabos DTE/DCE Cisco em suas interfaces seriais, e disposta numa área controlada sem qualquer outro tráfego presente nos enlaces.

As funções exercidas pelos componentes são as seguintes: as máquinas (Estação 1, Estação 2 e Estação 3) são fonte e destino dos pacotes gerados e rodam os softwares de geração, medição e visualização de tráfego. É nelas que são feitos os cálculos e as medições de tráfego, com base nos valores de QoS de interesse (*delays, jitters*, taxas de perdas e vazão).

---

1 Distribuição Conectiva 9.0, kernel 2.4.21

2 Modelo 2600 com IOS 12.1(5)T

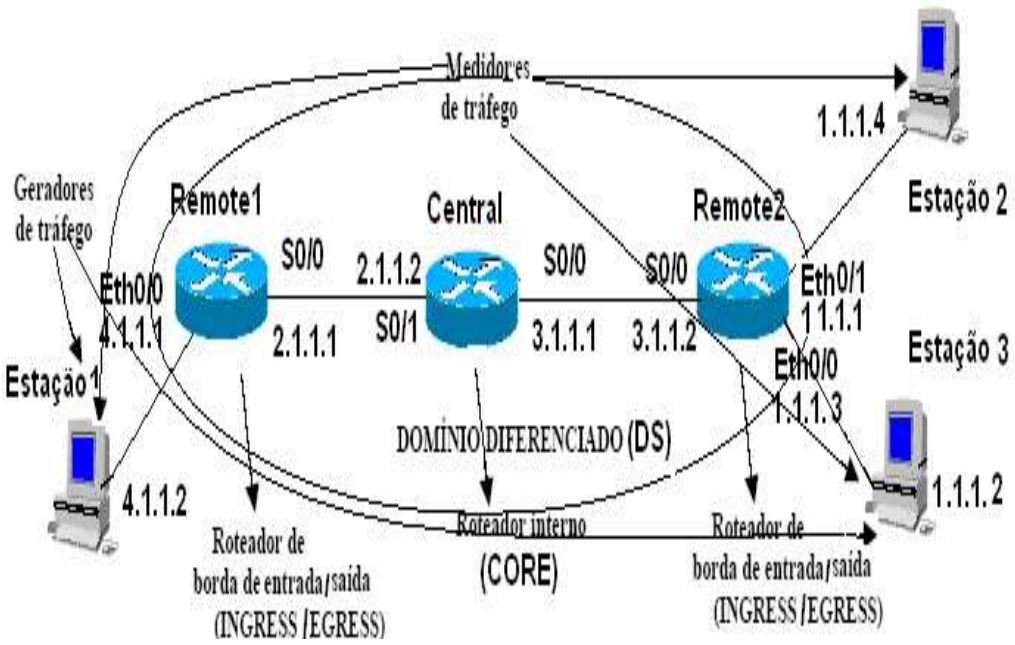


Figura 6-5 - Ambiente de testes para implementação do *DiffServ*

6.4 Metodologia adotada

As métricas de QoS adotadas foram o *delay* e sua variação (*jitter*), taxa de transferência e taxas de perda de pacotes. Foram utilizados, também, a quantidade de transações por segundo, tendo como parâmetro o tamanho dos pacotes enviados/recebidos e o tempo de duração de cada experimento.

Foi usado para medição de tráfego UDP a família MGEN [60] (*mgen*, *drec* e *mcalc),* respectivamente, para geração, recepção e medição dos tráfegos UDP. O MGEN permite a geração controlada de fluxos UDP, sendo possível a transmissão de tráfego *unicas*t ou *multicast*, com a taxa de bits desejada. A ferramenta permite ainda controlar o tempo do experimento e a variação do tamanho dos pacotes UDP estudados.

Para medição do tráfego TCP, foi utilizado a ferramenta Netperf [62], que faz a geração controlada do tráfego e a sua recepção. Esta ferramenta foi desenvolvida pela divisão de redes da *Hewlett-Packard Company* para medir aspectos de desempenho de rede. O princípio do Netperf é saturar o meio e retornar a vazão alcançada pelo fluxo analisado. Este *software* é composto

pelos programas *netperf ,* que gera tráfego a partir da estação transmissora, e *netserver,* que reflete tráfego o tráfego para a estação transmissora.

Esta ferramenta trabalha com uma estrutura cliente/servidor (receptor/transmissor) e permite medir a taxa de transferência unidirecional e a latência fim-a-fim. Com esta ferramenta podem ser feitas as medidas de tráfego *TCP Stream, UDP stream, TCP request/response e UDP request/response*.

A análise de rede foi realizada através da ferramenta *Netpeek,* que vem junto com a distribuição Linux e tem a função de mostrar o conteúdo dos pacotes que trafegam na rede. *O Sniffer Portable 4.7* [63], permitiu analisar o tráfego da rede da Previdência, o que forneceu os parâmetros para montagem do ambiente de teste, sendo, também, parte importante para o acompanhamento preciso dos pacotes durante os experimentos.

As políticas adotadas foram baseadas nas características das principais aplicações utilizadas pela Previdência Social e objetivou extrair das experiências, as configurações que obtivessem o melhor desempenho da rede. A variação dos testes também contemplou as particularidades de cada rede, pois nem todos os postos de atendimento possuem as mesmas funções nem atendem um só tipo de cliente.

6.5 Configuração dos roteadores

Nas configurações dos roteadores foram usadas *Access Control Lists* (ACL) para organizar os fluxos que entram no domínio *DiffServ*. Esta organização em lista de controle serviu como parâmetro na agregação dos fluxos e sua posterior classificação. As ACLs foram configuradas utilizando portas de origem/destino para fluxos UDP e endereços IPs para fluxos TCP, possibilitando, assim, determinar os fluxos correspondentes aos identificados na rede de produção, o que possibilitou simular os diversos fluxos necessários ao experimento. Nesta configuração foram utilizados PHBs conforme descrito na tabela 6-2.

Os mecanismos de moldagem de tráfego e policiamento foram inseridos

nas classes de menor prioridade para diminuir a influência destas classes no momento de maior concorrência. Foi introduzido na classe *Bronze* o mecanismo de QoS para moldagem de trafego (*Traffic Shape*), e na classe *Best-effot* o mecanismo de policiamento de tráfego (*Policing*), isto foi feito para que fosse verificada a eficácia destes mecanismos e analisado a empregabilidade no ambiente real.

Antes de qualquer teste foi necessário sincronizar os relógios das estações geradoras e receptoras. Isto foi conseguido através da ferramenta *ntpdate* presente na distribuição Linux utilizada. A precisão requerida do tempo em que os pacotes percorrem de um lado a outro da rede (*one-way delay*) é um parâmetro crítico e requer cuidados para que as medidas sejam confiáveis. Isto devido ao cálculo realizado pelas ferramentas de medição que calculam o *delay* tomando como base a diferença numérica entre o tempo em que o pacote chega no receptor e o tempo em que o mesmo é transmitido. Uma diferença em milisegundos no sincronismo acarreta um erro significante ao final do processo de medição. Para evitar problemas desta natureza, fez-se um processo automático de sincronização a cada medida nas máquinas do testbed.

Outro fato importante foi a necessidade introduzir relógios nos roteadores para que este sincronizem o tráfego entre roteadores de borda e centro, pois a comunicação (DTE/DCE) entre os mesmos é síncrona.

A configuração dos roteadores segue a classificação do tráfego e a codificação DSCP, conforme tabela abaixo:

Classe de Tráfego	**DSCP**	**PHB**	**Porta (Teste UDP)**	**ACL (Acces-group)**
Premium	46	EF	5001	101
Gold	10	AF1	5002	102
Silver	18	AF21	5006	108
	20	AF22	5007	109
	22	AF23	5008	110
Bronze	26	AF3	5003	104
Best-effort	0	BE	5005	105

Tabela 6-2 - Classes de tráfego

A estratégia de configuração dos testes obedeceu a critérios de similaridade com o ambiente da Previdência, onde encontramos variação de velocidades de 64 Kbps a 512 Kbps, sendo que a grande maioria das localidades estão conectadas na faixa de 64 Kbps a 128 Kbps. Devido à existência de uma grande variedade de aplicativos envolvendo diversas tecnologias, os pacotes que trafegam na rede tem tamanhos variados. Por este motivo foram feitas aproximações e ponderações para chegar aos valores de tamanho de pacotes para os testes. Neste sentido foram montados vários testes com diversas variações de parâmetros para abranger as principais situações de fluxos encontrados no ambiente em questão.

Para uma primeira série testes UDP, foram estabelecidos uma largura de banda em 64 Kbps, pacotes de 64 bytes, e a configuração das políticas para cada classe conforme abaixo:

Platinum – Política de QoS LLQ (PQ + CBWFQ) [61], prioridade estrita e largura de banda 32 Kbps (50%);

Gold - Política de QoS CBWFQ, política de descarte *Tail Drop*, prioridade alta e largura de banda 22,4 Kbps (35%);

Silver - Política de QoS CBWFQ, política de descarte *Tail Drop*, prioridade alta e largura de banda 16 Kbps (25%);

Bronze - Política de QoS CBWFQ, política de descarte *Tail Drop*, Prioridade alta e largura de banda 9,6 Kbps (15%), com *Traffic Shape* a 32 Kpbs;

BE - Sem política de QoS, enfileiramento WFQ, largura de banda 16 Kbps (25%) com policiamento de tráfego a 16 kbps, setando os pacotes não conformes com DSCP 0 e dropando o excedente.

A soma dos percentuais estabelecidos para cada banda é maior que 100%, mas não foram utilizadas, nestes testes, todas as classes simultaneamente. Foram usadas combinações que permitiram simular tráfegos dentro dos limites. Foram utilizadas em alguns testes bandas menores que 25% para a classe BE, o que não deve acontecer no ambiente de produção, mas que nos deram informações de comportamento do ambiente nos casos de congestionamento de todas as classes prioritárias.

Em um segundo teste UDP, foram modificados a largura de banda para 128 Kbps e tamanho de pacote para 128 bytes e foram mantidos todos os outros parâmetros.

Para marcar o esquema montado na tabela 6-1, foi usado o *Policy Map*[1], que foi denominado de SETDSCP, na entrada ethernet (ETH0/0) dos roteadores de borda (Remote1 e Remote2). A maneira que se declara uma política no roteador é a seguinte: Primeiro são montados as ACLs para o controle de admissão e agregação de fluxos, depois são relacionadas classes (*class-map*) com agregados estabelecidos pelas ACLs, em seguida são mapeadas as políticas (*police-map*).

Na interface de entrada dos roteadores de borda é feito, através da declaração das políticas, os relacionamentos das classes com seus respectivos

[1]Comando utilizado para declarar uma política no roteador

DSCP. Este procedimento faz a marcação do tráfego na entrada do domínio *DiffServ*.

Com o tráfego marcado na entrada com seus respectivos DSCP, os diferentes requerimentos das agregações são reunidas na configuração do *Policy Map.* Este *Police Map* é configurado na interface serial dos roteadores de borda na direção dos roteadores de centro. Nestas interfaces são estabelecidas as larguras de banda, policiamento, moldagem e políticas de descartes. É com ajustes nestes parâmetros que chegamos a montar uma boa política.

Esta configuração é a estabelecida para os testes UDP e TCP, tendo como algoritmo de policiamento o *Token Bucket* e MTU com tamanho de 1500 (*default*). Nesta configuração foram usadas para montagem das ACLs portas de origem/destino para fluxos UDP e endereços IPs para fluxos TCP. Assim foram determinados os agregados de fluxos que correspondem aos fluxos reais. Isto supriu as necessidades de laboratório e possibilitou simular os diversos fluxos do experimento. Foram introduzidos, na classe *Bronze,* os mecanismos de QoS para moldagem de trafego (*Traffic Shape*), e na classe *Best-effot* o mecanismo de policiamento de tráfego (*Policing*). Isto foi feito para que fossem verificada a eficácia destes mecanismos e feito a análise da empregabilidade no ambiente real.

Nos roteadores de centro não colocamos políticas para fluxo de entrada, já que o fluxo que entra neste roteador já vem marcado pelos roteadores de borda. Configuramos as interfaces de saída para fazer a classificação e aplicação de políticas de QoS em ambas as direções, com WRED para controle de tráfego nas classes *Gold*, *Silver* e *Bronze*. A classe *Platinum* foi configurada com política de QoS LLQ (PQ + CBWFQ) [61].

Os testes foram feitos variando fluxos cargas e os fluxos em background, de um limite abaixo da banda reservada até uma situação de sobrecarga.

6.6 Testes UDP

O teste de simulação com tráfego UDP envolveu a configuração dos softwares geradores e receptores, que através das portas, injetaram e recepcionaram sinais, conforme tabela 6-1, e foram especificados conforme abaixo:

Geração de tráfego com fluxos individuais com as classes *Platinum*, *Gold*, *Silver*, *Bronze* e BE;

Tráfego *Platinum* em concorrência com cada classe, variando a carga *Platinum* e a concorrente;

Tráfego *Gold* em concorrência com cada classe, variando a carga *Gold* e a concorrente;

Tráfego *Gold* em concorrência com a classe *Silver* e *Bronze*, variando a carga *Gold* e as concorrentes. Repetimos este teste para cada classe, tendo o cuidado de não excedermos os percentuais que estabelecemos para cada classe, isto é, não poderíamos ter tráfego para todas as classes porque a soma dos percentuais excederia a banda, causando uma situação irreal. As situações de sobrecarga são conseguidas pelo aumento controlado de carga para cada classe referente as suas reservas de banda.

6.7 Testes TCP

Para os testes TCP foram usados os endereços IP origem e destino, tendo o cuidado de envolver todos os fluxos da conexão observada para que os mesmos tenham a mesma prioridade e não influam negativamente na desempenho do experimento. Todos os testes TCP envolveram fluxos em ambos os sentidos, sendo controlado o tamanho do pacote enviado e tamanho do pacote de resposta conforme observado na caracterização do tráfego prioritário.

Neste momento do desenvolvimento do trabalho houve uma aumento na largura de banda dos enlaces da Previdência de 64 kbps para 128 kbps. Esta

situação conduziu a mudanças nos parâmetros de testes, mas antes de serem efetivadas as mudanças, foram verificados, com simulações, o comportamento das testes anteriores em relação a uma banda de 128 kbps e não houve alteração significante. Então, para os testes TCP foi modifica a configuração das políticas, sendo atribuída uma largura de banda de 128 Kbps, permanecendo com as mesmas classes, mas modificados os percentuais para cada classe, conforme abaixo:

Platinum – Política de QoS LLQ (PQ + CBWFQ), prioridade estrita com tráfego crítico garantido e largura de banda 19,2 Kbps (15%);

Gold - Política de QoS CBWFQ, política de descarte *Tail Drop*, banda garantida para classe de tráfego definida e largura de banda 25,6 Kbps (20%);

Silver - Política de QoS CBWFQ, política de descarte *Tail Drop*, banda garantida para classe de tráfego definida e largura de banda 25,6 Kbps (20%);

Bronze - Política de QoS CBWFQ, política de descarte *Tail Drop*, banda garantida para classe de tráfego definida e largura de banda 25,6 Kbps (20%);

BE Sem política de QoS, enfileiramento WFQ, largura de banda 32 Kbps (25%).

A diferença entre as classes *Gold*, *Silver* e *Bronze* está nos níveis de precedência de descarte, onde a classe *Gold* tem menor probabilidade de descarte em relação à *Silver* e esta à *Bronze*.

Este primeiro conjunto de testes TCP foi feito em duas etapas, uma com 1, 4 e 6 fluxos TCP concorrentes sem QoS e outra com os mesmos fluxos TCP com QoS. Tiveram, ambos, as seguintes variações de cargas em background: Sem carga, com fluxos UDP e com fluxos UDP+TCP. Na presença de QoS, os testes com 6 fluxos TCP foram compostos de 4 fluxos classificados e 2 fluxos BE.

Para o segundo conjunto de testes TCP, foram modificadas as políticas das classes *Gold*, *Silver* e *Bronze* para CBWFQ com política de descarte WRED. Estas configurações foram submetidas aos mesmos parâmetros do primeiro teste.

6.8 Considerações sobre Este Capítulo

Este capítulo descreveu toda montagem do ambiente de teste, envolvendo tanto a parte de hardware como de software. Tratou da instalação e configuração dos softwares, da metodologia de medição adotada e aplicação nos diversos testes, além dos registros das medições. A seguir, no próximo capítulo, serão analisados e relatados os resultados dos testes montados neste capítulo.

7 MEDIÇÕES, ANÁLISES E RESUTADOS

Baseado nas configurações estabelecidas no capítulo anterior, este capítulo mostra as medições e as análises de cada experimento e expõe, através de gráficos e comentários, os resultados obtidos. É observado o comportamento de cada experimento, registrado e analisado as variações de desempenho, feito recomendações e restrições de uso. Abaixo estão descritos os objetivos, os registro dos dados, a demonstração gráfica e as análises individualizadas de cada teste.

7.1 Teste 1 – Implementação de QoS sem carga concorrente

O objetivo deste teste é verificar o comportamento das políticas de QoS na ausência de concorrência em cada classe separadamente e o comportamento do *shape average* configurado na classe *Bronze*. O teste utiliza a configuração básica estabelecida para os primeiros testes UDP, conforme descrito no capítulo anterior.

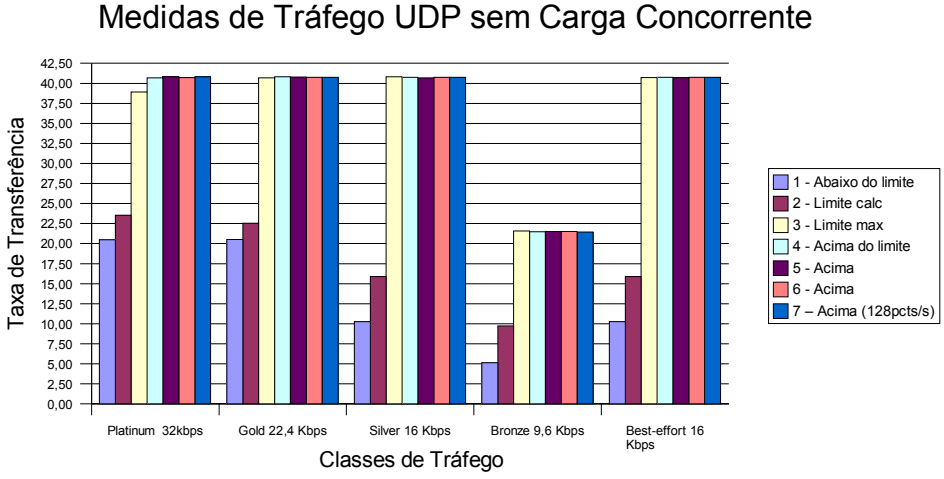
Um teste desta natureza serviu para mostrar o comportamento dos mecanismos de QoS na ausência de concorrência. Os testes submeteram o DS a uma carga que aumentava gradativamente até chegar o ponto máximo da taxa de transferência e posterior saturação dos meios de transmissão. O gráfico da figura 7-1 mostra o comportamento destes testes.

As informações mais importantes neste teste foram os dados relativos ao comportamento de cada classe nos momentos críticos de ocupação da banda. A classe *Platinum* não variou a latência, mas descartou todo o tráfego que excedeu o limite da banda. As classes *Gold* e *Silver* tiveram comportamentos idênticos, ocuparam toda banda disponível e conseguiram transmitir mais pacotes que a *Platinum*, mas a um custo do aumento considerável da latência. A classe *Bronze* teve seu tráfego máximo limitado à metade da largura disponível devido às restrições imposta pelo *shape average*, o que foi obedecido prontamente, e descartou todo o tráfego excedente. A classe BE, sem QoS, começou a descartar logo ao atingir a largura máxima disponível, a

uma taxa maior que as classes *Gold* e *Silver*, e menor que a classe *Platinum*. Teve latência menor que as classes *Gold* e *Silver*, e maior que a classe *Platinum*.

Conforme figura 7-1, podemos verificar que neste teste todas as classes excederam sua faixa e ocuparam toda banda disponível, exceto a classe *Bronze*, que foi configurando com *shape average*. O *shape* foi obedecido prontamente e esta classe excedeu até o limite do *shape*. Os mecanismos de QoS agiram de maneira precisa na classe *Platinum* e no *shape average* da classe Bronze, não sendo possível observar com detalhe as classes *Gold* e *Silver* devido sua similaridade, pois sua diferença está só nas priorizações de descartes, o que é muito sutil e difícil de ser observado com os recursos utilizados.

Figura 7-1 – Tráfego UDP Sem Concorrência



Neste teste podemos concluir que para situações onde a necessidade é priorizar apenas um tipo de tráfego, pode-se utilizar qualquer das políticas testadas, mas se é conhecido o comportamento do fluxo e a necessidade de QoS da aplicação, pode-se escolher mais adequadamente os mecanismos de QoS aplicáveis. Por exemplo, para os casos onde a aplicação necessita de priorização absoluta e utiliza largura de banda conhecida e estreita, é indicado utilizar a classe *Platinum*, pois esta classe garante execução sem perdas e com

latência mínima quando as taxas de transferência estão iguais ou menores que a largura disponível para a classe. Sempre observando o comprometimento da banda com este tipo de tratamento, pois pode haver uma degradação de desempenho das outras aplicações.

Os dados deste testes não são conclusivos para utilização concorrentes de várias classes, pois foram testados apenas tráfego UDP, mas são cumulativos para posterior análise.

7.2 Teste 2 - Classe *Platinum*

Este teste objetiva observar o comportamento da classe *Platinum* submetida a concorrência da classe *Gold*. Os teste foram feitos submetendo-se uma carga *Platinum* a uma variação de carga *Gold* concorrente. A variação da carga Gold foi iniciada com valores abaixo de sua faixa reservada até exceder a largura máxima da banda. Depois desta primeira medida, foi repetido para mais seis valores de carga *Platinum*. Os valores da carga *Platinum* variaram de uma carga dentro da banda reservada até um valor de transbordo da largura máxima de banda.

A classe *Platinum* teve o mesmo comportamento em todos os testes nas diversas variações das carga concorrente, mesmo onde o trafego estava acima da banda disponível. Não houve variações significantes de latência e os pacotes que excederam a capacidade da banda disponível para a classe, foram descartados incondicionalmente, conforme o esperado para esta política. O gráfico da figura 7-2 mostra o comportamento da classe *Platinum* com a concorrência de diversas cargas da classe *Gold*. Neste gráfico verificamos que a classe *Platinum* permanece com a mesma taxa de transferência em qualquer variação de carga concorrente (linhas contínuas superpostas). *A*s perdas de pacotes[1] (linhas pontilhadas superpostas) aumenta nas mesmas proporções do aumento da carga *Platinum*, independente da variação da carga concorrente, isto devido ao compromisso de manter uma latência baixa. Esta classe não faz

[1]As taxas de perdas correspondem à quantidade de pacotes descartados por segundo no roteador de borda representado pelo seu equivalente em $K_{bps.}$

tratamento antes do descarte de pacotes.

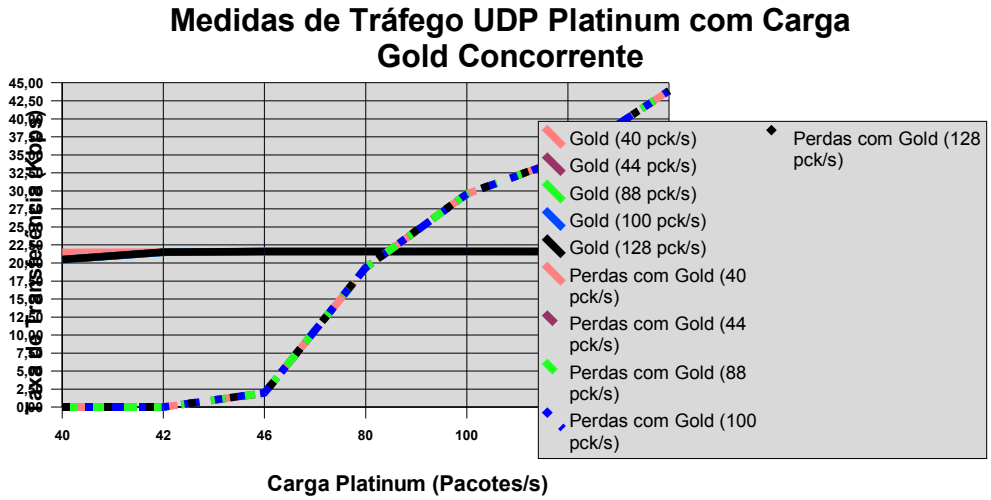


Figura 7-2 – Tráfego UDP Com Classe Platinum

Estes testes são conclusivos quanto a eficácia da classes *Platinum*, sua independência de interferência de outras classes e prioridades, sua eficácia na manutenção de latência mínima e sua política de descarte sem ponderações de prioridade, além de utilizar banda disponível quando a concorrência é baixa. Estes testes confirmaram que sua aplicabilidade deve ser criteriosa, uma vez que pode prejudicar o desempenho da rede. É uma política indicada para situações onde, além da necessidade de prioridade absoluta, os fluxos tenham comportamentos previsíveis. Adequado para tráfego de voz e CBR, se não, ele deve ser bastante largo para absorver *burst*.

7.3 Teste 3 - Classes *Platinum*, *Gold, Silver, Bronze e BE*

O objetivo deste teste é verificar o comportamento das classes *Platinum*, *Gold, Silver, Bronze e BE* concorrendo entre si. Nestes testes são simuladas situações de concorrência entre todas as classes em diversas situações. Cada medida compreende um aumento progressivo de carga (pacotes/s) de todas as classes, sendo que a classe BE permanece com uma carga constante de 22 pacotes/s. No gráfico da figura 7-3[2], temos o resultado dos testes simultâneos

[2]Gráfico mostrado em barras devido a coincidências dos valores que acarretaria numa coincidência de linhas

de todas estas classes. Nestes testes podemos verificar que a classe *Platinum* não sofre alteração em nenhum momento de congestionamento, mas as classes *Gold*, *Silver* e *Bronze*, na hora de congestionamento, ocupam toda banda garantida e invadem a banda reservada à classe BE, tornando-a inviável. Isto é notado na coincidência das curvas *Gold, Silver* e *Bronze* em todos os testes, que aumenta a taxa de transferência com o declínio da classe BE até um valor mínimo de utilização.

Figura 7-3 – Classes Platinum, Gold, Silver, Bronze e BE Simultâneas.

Neste teste podemos concluir que as classes *Gold, Silver e Bronze* podem trabalhar em conjunto tendo-se o cuidado de não deixar tráfegos importantes na classe BE. É importante ter sob controle o fato de todos os tráfegos priorizados não estarem simultaneamente presentes com carga máxima ao mesmo instante, pois isto inviabiliza o tráfego não priorizado. O mais interessante para enlaces de banda estreita é priorizar poucos fluxos que ocupem uma pequena fração da banda e dar um tratamento igual ao restante dos fluxos. Pode-se também lançar mão do *shape* para garantir que tráfego menos importante ocupem demasiadamente a banda ou, que embora importante, não necessite de garantias restrita de QoS.

7.4 Teste 4 - Classe *Gold*

O objetivo deste teste é mostrar a variação da taxa de transferência da classe *Gold* nas condições de concorrência com a classe *Silver* em momentos distintos, combinado com as correspondentes perdas sofridas. Este teste serve para mostrar o aproveitamento da banda por duas classes priorizadas, onde ambas tem a flexibilidade de usar toda banda disponível. Para melhor entendimento do gráfico mostrado na figura 7-4, as curvas de perdas retratam uma correspondência em Kbps da quantidade de pacotes descartados na interface de entrada do domínio DS. Pela análise deste gráfico verifica-se que a melhor situação observada é quando temos a carga S*ilver* está a 20 pacotes/s (abaixo do limite máximo) e *Gold* entre 40 e 64 pacotes/s. Neste trecho temos aproveitamento máximo da classe *Gold* sem perda. Também, podemos perceber que esta classe está acima do limite máximo e sem perda, isto porque neste instante esta classe esta utilizando banda não utilizada pelas outras classes. A situação inversa também tem o mesmo desfecho, pois estas classes tem comportamentos semelhantes.

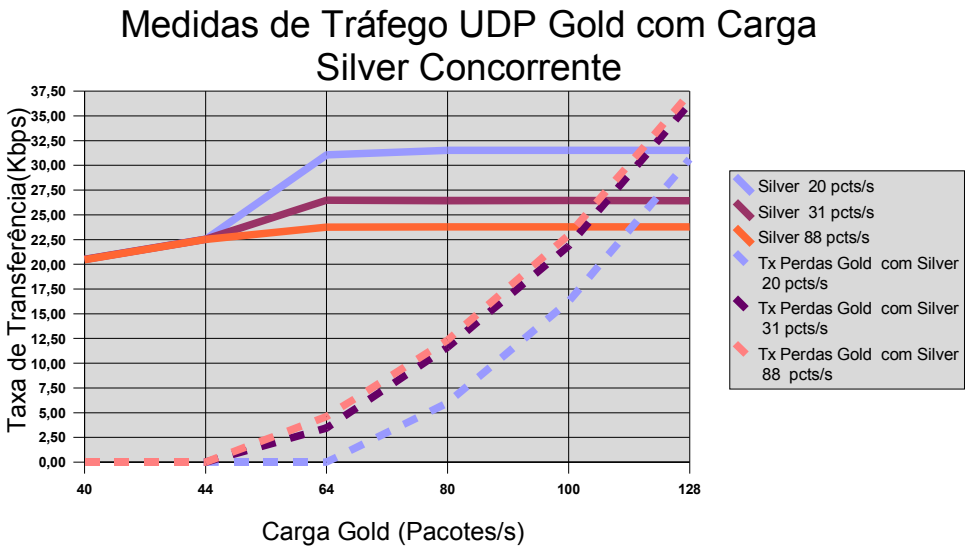


Figura 7-4 – Classe *Gold* com Concorrência Silver

Notamos que há uma interação entre estas classes, quando há banda disponível a classe que necessita de banda aproveita esta disponibilidade mas

não interfere na banda garantida para a outra classe concorrente.

7.5 Teste 5 - Latência e perdas de pacotes com classes *Gold, Silver e Bronze*

Este teste tem o objetivo de mostrar a variação da latência e as perdas de pacotes das cargas *Gold, Silver e Bronze* submetidas a sucessivos aumento de carga e concorrência. A utilização apenas de três classes deve-se ao fato da classe *Platinum* já ter comportamento conhecido e a classe BE não ter prioridade.

O gráfico da figura 7-5 mostra o aumento da latência com o aumento da carga para as classes *Gold, Silver* e *Bronze*. Nestes valores podemos observar que as classes *Gold e Silver* têm o mesmo comportamento, sofrendo um crescente aumento de latência com o aumento da carga, mas a classe *Bronze* tem uma curva de latência mais branda devido a política de conformação de tráfego presente em sua configuração. Os mecanismos *Traffic* Shape (GTS) baseado em classes atrasam os pacotes usando algoritmo WFQ e descartam o excedente. Mas, para manter esta curva, este algoritmo usa um mecanismo de descarte na interface de entrada mais rigoroso que os mecanismos das classes *Gold e Silver*. Verificamos, na figura 7-6, que o descarte de pacotes excedentes da classe *Bronze* inicia antes das outras classes, já que a banda é estreitada pelo *shape*, mas a latência varia de maneira mais branda devido à política estabelecida, que para manter a latência estável (figura 7-6) esta política promove um acentuado descarte de pacotes.

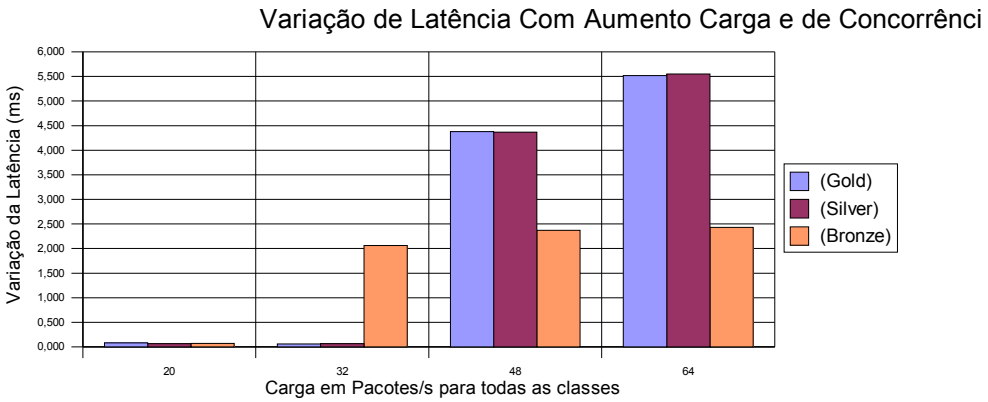


Figura 7-5 – Variação de Latência x Classe

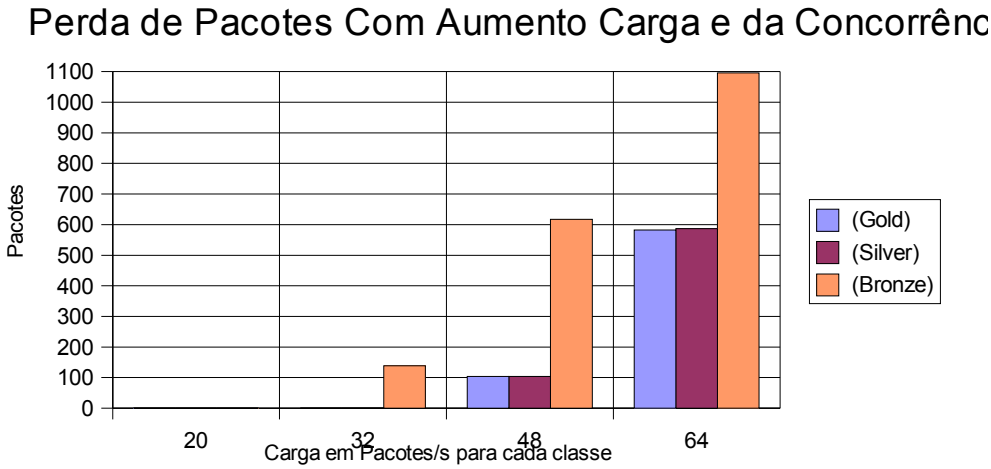


Figura 7-6 – Perdas x classe

Pelo que foi observado estas classes tem comportamentos semelhantes e não são adequadas para serem utilizadas em fluxos sensíveis a perdas sob condições de concorrência pesada. Devido ao aproveitamento dinâmico da banda, estas classes são mais indicadas para priorizarem tráfegos não sensíveis a atrasos e sob concorrência branda. Nesta situação pode haver ligeiro aumento de latência, perdas mínimas de pacotes e máxima utilização de banda.

Os testes acima envolveram apenas tráfego UDP, abaixo são mostrados

os testes envolvendo tráfego TCP e UDP concorrentes. As características do tráfego TCP são diferentes do tráfego UDP, pois envolve confirmação e podem ocorrer retransmissões, por este motivo os testes TCP necessitam do envolvimento de tráfego bilateral.

7.6 **Teste 6 - Classe *Gold* TCP**

Este teste tem o objetivo de mostrar o comportamento da classe *Gold* com carga TCP, utilizando a política CBWFQ e descarte *tail drop*, com variação de concorrência UDP *Platinum, Silver e Bronze*. Esta é uma situação em que é estabelecido uma carga TCP para envio e outra para retorno, depois se submete estes fluxos a uma concorrência. Repete-se estas medições para várias variações de cargas TCP. A transposição destes dados gera uma curva, conforme no gráfico da figura 7-7. As outras curvas são conseqüências das mesmas variações de cargas TCP com variações de concorrência UDP, também refletido no gráfica da figura 7-7.

Este teste mostrou que em todos os experimentos houve uma variação significante de desempenho devido ao tamanho dos pacotes que circulam em ambos os sentidos da conexão, como podemos observar no gráfico da figura 7-7, e nos demais experimentos que foram feitos com fluxo TCP. Isto devido as característica do protocolo TCP, como mostrado no item 6.1.2. Neste gráfico está representado a variação de tamanho de pacotes enviados/recebidos e sua conseqüente taxa de transferência. É observado que quando a concorrência é muito pesada o desempenho cai ao limite de banda garantida à classe, como observado na curva que representa a carga concorrente (pla=30, sil=bro=50 pct/s), que é uma carga bastante pesada para a banda disponível. Nesta situação de concorrência a banda reservada à classe BE é utilizada pelos fluxos UDP das classes *Silver* e *Bronze* concorrentes e pelo fluxo TCP submetido à políticas de controle de congestionamento e à própria característica TCP. As variações das curvas com o tamanho dos pacotes enviados e recebidos está ligado às características do TCP. É visto que há uma otimização do uso de banda quando as cargas TCP estão entre 512 e 1024

bytes, isto devido aos mecanismo do TCP conseguirem uma utilização com o máximo de aproveitamento de banda.

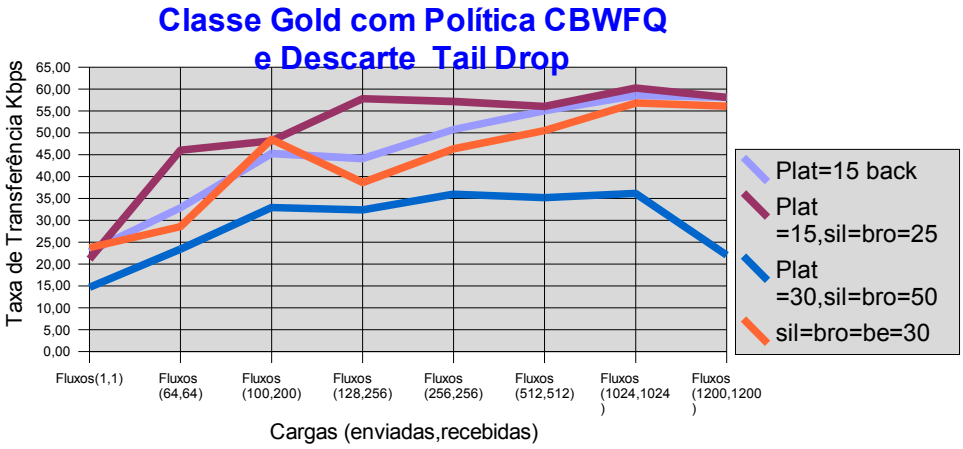


Figura 7-7 – CBWFQ com *Tail Drop* variando carga e concorrência

Pelo que foi observado, este teste mostra o efeito da concorrência sobre o desempenho de uma classe priorizada, quanto a sua taxa de transferência, mas não são conclusivos quanto ao tempo de resposta das aplicações. Os tempos de respostas estão ligados tanto à concorrência quanto ao tamanho dos pacotes nos dois sentidos. O fluxo real possui tamanhos diferentes de pacotes em ambos os sentidos, e este fluxo adequa-se a largura de banda disponível conforme mecanismo de janelas de tamanho variável do TCP. Por este motivo, ao escolher tráfego a priorizar, esta política de descarte pode não ser eficiente quando a largura disponível é estreitada.

7.7 Teste 7 – Classe Gold TCP e carga UDP em background para política de descarte WRED e *Tail Drop*

O objetivo deste teste é observar o comportamento da classe *gold* com política CBWFQ nas situações de testes sem política de descarte, com políticas de descartes *tail drop e* com políticas de descartes WRED. Também, verificar a influência destas políticas sobre as perdas de pacotes das classes UDPs concorrentes.

Na figura 7-8, é mostrado um conjunto de curvas representando a taxa de

transferência da classe *Gold* com alta concorrência *Silver*, *Bronze* e BE(linha contínua) e a correspondente perda de pacotes (linha pontilhada). Neste gráfico podemos verificar que há um crescimento de aproveitamento de banda com o aumento do tamanho dos pacotes até um valor entre 80 e 120 kbps, conforme política adotada, que depois sofre um declínio com fluxos maiores que (1200, 1200) bytes. Este fenômeno acontece devido às características do tráfego TCP, pacotes pequenos requerem mais fluxos de controle acarretando perda no aproveitamento de banda. Pacotes muito grandes em banda estreita sofre problemas no dimensionamento dinâmico das janelas deslizantes e, também, deteriora o aproveitamento da banda. A otimização do uso da banda se dá quando há um equilíbrio entre tamanho de pacotes e banda disponível onde o mecanismo TCP consegue uma significante melhora de desempenho.

As perdas são mais acentuadas quando não aplicamos nenhuma política, isto porque não há tratamento de tráfego, os pacotes que excedem são imediatamente descartados, já as outras duas políticas, se diferenciam no tratamento que a WRED faz antes de descartar, esta política verifica a precedência e descarta com base nos DSCPs, isto diminui o aproveitamento da banda mas torna a priorização de descarte mais eficiente. O fluxo TCP sem QoS e o fluxo com política de descarte *tail drop* são mais irregulares devido a política de descarte ser menos eficiente, o que pode provocar retransmissão de tráfego prioritário.

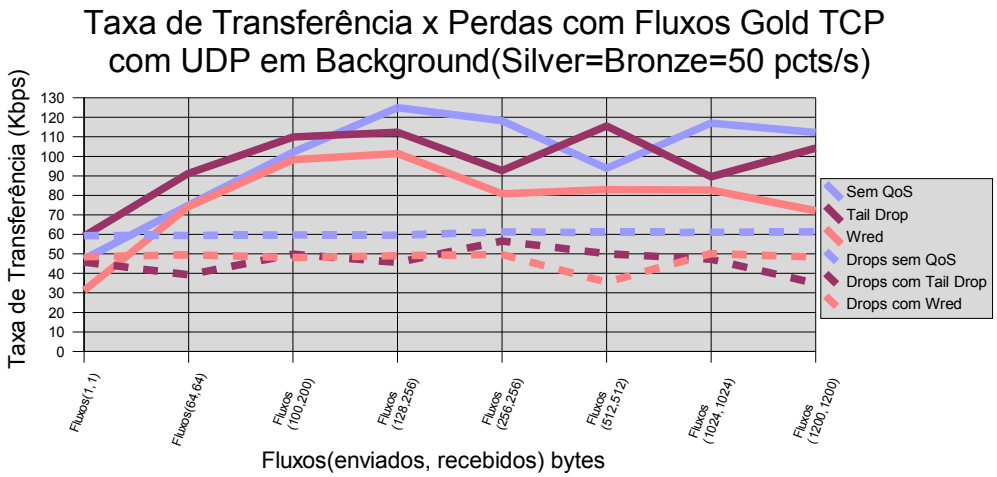


Figura 7-8 – Taxa de Transferência x Perdas com Fluxos *Gold* TCP

Para aplicações sensíveis a perdas mas tolerantes a atrasos é conveniente utilizar uma política de descarte WRED, pois o tráfego priorizado é tratado de maneira diferenciada e o descarte é baseado no DSCP.

7.8 Outras considerações

As classes *Gold*, *Silver*, *Bronze* e BE aumentam sua latência nos momentos de congestionamento. Em todas as classes a latência aumenta abruptamente no limiar do descarte, exceto a classe *Platinum*, que sofre variações de latência muito discreta em qualquer situação.

Nos testes TCP com concorrência UDP ou UDP + TCP, a banda reservada para cada classe é melhor utilizada na concorrência de vários fluxos por classe, até uma determinada quantidade de fluxo. A variação no tamanho dos pacotes influencia na aproveitamento banda, o desempenho aumenta até um determinado tamanho de pacote depois começa a perder desempenho. Assim podemos dizer que o tamanho dos pacotes e o número de fluxos em uma classe são fatores determinantes na eficácia da utilização de banda.

A presença de vários fluxos concorrentes melhora a utilização da banda, mas compromete a previsibilidade, pois as filas que estes fluxos obedecem dentro das classes podem ter algoritmos do tipo *fifo*.

Na presença de vários fluxos em uma classe, a ordem de entrada tem influencia sobre seu desempenho, isto é, vários fluxos idênticos no momento de congestionamento, têm desempenho distintos, dependendo da ordem de entrada.

A concorrência entre classes é bem previsível, mas quando temos um grande número de fluxos concorrendo dentro de uma classe é conveniente conhecer bem o mecanismo que vai fazer o tratamento destes fluxos, para tornar suas execuções previsíveis. A observação da largura de banda para comportar os fluxos simultâneos é um item de projeto importante e deve ser bem dimensionado.

É verificado que, sob concorrência pesada, tráfego TCP não classificado enfileira e pode não rodar ou ter uma desempenho impraticável, quando utilizamos CBWFQ.

7.9 Considerações sobre Este Capítulo

Neste capítulo foram relatados os resultados obtidos em testes e as análises correspondentes aos dados extraídos, fazendo considerações em relação à proposta de implementação no ambiente real. Detalhes sobre os estudos experimentais também foram descritos neste capítulo.

8 CONCLUSÕES

A rede da Previdência é composta por enlaces de baixa velocidade, que proporciona uma taxa de transferência baixa, e equipamentos (roteadores) com IOS Cisco com suporte a QoS. A população servida pelos serviços da Previdência é composta por pessoas simples, pessoas que se deslocam com dificuldade, deficientes físicos, gestantes, aposentados e trabalhadores afastados de suas atividades devido a problemas de saúde. Nada mais justo que priorizar os serviços voltados a esta população e racionalizar os outros serviços nos momentos de congestionamento.

Neste trabalho observamos que todas as redes da Previdência se beneficiam com a implementação de QoS da Cisco. Largura de banda, latência e perda de pacotes podem ser controladas com eficiência. Garantindo os resultados, os mecanismos de QoS conduzem a serviços eficientes e previsíveis para aplicações vitais ao negócio da empresa. A utilização de QoS permitirá a oferta de gerenciamento, banda por demanda entre outros serviços disponíveis. A favor desta implementação esta a plataforma formada por roteadores que suportam as funcionalidades de QoS existente em toda rede da Previdência. As questões principais a serem analisadas para a implantação de qualidade de serviço na Previdência rondam em torno de que percentual de canal disponibilizar para os serviços propostos. Conforme observado na caracterização de tráfego, a necessidade de banda para os serviços essenciais é função da quantidade de pontos de atendimento e da política de QoS adotada. Se houver priorização absoluta, deve haver, além da largura calculada, uma margem de segurança para que não haja descarte de pacotes. Se a opção for uma priorização mais moderada a preocupação deve ser com o tráfego não priorizado, pois este pode não ser suprido. Após o conhecimento disto é possível indicar, pelo menos, um percentual inicial que atenda parte desta demanda e qual o impacto desta alocação para as aplicações convencionais da rede. Como os roteadores da infra-estrutura montada suportam as

funcionalidades de QoS pode-se implementar QoS IP em toda a rede sem a necessidade de compra de roteadores novos, devendo apenas atualizar versões de IOS e implementação de memória.

O objetivo deste trabalho foi alcançado, uma vez que conseguimos demonstrar o ganho com a implementação de QoS numa rede semelhante à rede da Previdência Social. A plataforma Cisco foi muito eficaz na implementação de QoS nos roteadores de borda, o Policiamento e o *Shaping* funcionaram perfeitamente, as perdas foram diminuídas com aplicação de política de descartes e a racionalidade no uso da banda foi conseguido através da estratificação por classes combinado com os mecanismos de QoS, o componente de centro foi configurado de maneira simples e funcionou perfeitamente dentro do especificado.

Um aspecto negativo desta plataforma foi a impossibilidade de diferenciação da política de descarte entre as classes *Gold* e *Silver,* nos diversos experimentos. Estas classes tiveram comportamentos semelhantes em todos os testes, embora fosse esperado um descarte diferenciado, porém as ferramentas utilizadas não captaram dados que as diferenciasse.

Pelos testes feitos é possível concluir que pela simples implementação de QoS na plataforma existente a previsibilidade dos fluxos classificados é aumentada e a garantia de sua execução é conseguida. Todas as políticas implementadas deram estas certezas, as variações experimentadas mostraram com mais clareza as diversas características que podem ser aproveitadas para conformar os fluxos de maneira estratificada por prioridades.

Estes resultados demonstram a eficácia da arquitetura de Serviços Diferenciados ao permitir a coexistência de fluxos de várias naturezas, mesmo em condições de congestionamento extremo. Esta característica torna esta solução bastante viável para os serviços prestados pela Previdência Social.

8.1 Trabalhos Futuros

Com a previsão de utilização de voz sobre IP (VOIP) dentro da rede da Previdência e um possível aumento de tráfego de tempo real, é interessante a ampliação deste estudo para mecanismos de eficiência de links. Uma maneira é a fragmentação e intercalação de pacotes, que pode melhorar esta comunicação fragmentando pacotes grandes e intercalando no fluxo diminuindo a espera na fila de saída. Outra maneira de melhorar a eficiência é diminuindo a carga comprimindo o cabeçalho através do *Compressed Real-Time Protocol header*.

No desenvolvimento deste trabalho percebeu-se a necessidade de ajustes na rede para a obtenção de bons resultados, neste sentido, um estudo envolvendo Engenharia de Tráfego, que já é atualmente uma função indispensável em grandes redes por causa do custo alto dos equipamentos e da natureza comercial e competitiva da Internet, seria bastante adequado para melhor utilização dos mecanismos de controle de tráfego. A Engenharia de Tráfego altera o fluxo normal dos pacotes, assim, ela pode ser utilizada para atender a requisitos de QoS de determinados fluxos de dados. Um trabalho desta natureza pode ser feito manualmente, ou usando algum tipo de técnica automatizada, usando inclusive MPLS ou Roteamento com QoS para descobrir e fixar os caminhos mais adequados a determinados fluxos dentro da rede.

Outro estudo interessante seria a utilização combinada de tecnologias como: *DiffServ* com *IntServ* aproveitando as característica do *IntServ* para os nós de fronteira e *DiffServ* para os nós de centro, por exemplo, ou Utilizar MPLS com *DiffServ* aproveitando a similaridade entre ambos ou MPLS com IntServ.

REFERÊNCIAS BIBLIOGRÁFICAS

[1] Braden, R., Clark, D. & Shenker, S., I*ntegrated Services in the Internet Architecture: an Overview*, RFC 1633, Junho 1994.

[2] Blake, S., Black, D. L., Carlson, M., Davies, E., Wang, Z. e Weiss, W. (1998). *An Architecture for Differentiated Services,* Internet RFC 2475

[3] Rosen, E. et al., *Multiprotocol Label Switching Architecture*, Internet Draft, Agosto 1999.

[4] Crawley, E. et al., *A Framework for QoS-based Routing in the Internet*, RFC 2386, Agosto 1998.

[5] Joberto Martins, Qualidade de Serviço (QoS) em Redes IP Princípios Básicos, Parâmetros e Mecanismos, JSMNet Networking Reviews - Vol. 1 - N° 1

[6] Andrew Tanembaum, "C*omputer Networks*", 3rd edition, Prentice-Hall, 1996

[7] Adailton J. S. Silva, Qualidade de Serviço em VOIP, NewsGeneration, 12/05/2000, vol 4, numero 3

[8] Melo, Edson Tadeu Lopes, Qualidade de Serviço em Redes IP com *DiffServ*: Avaliação através de Medidas, Universidade Federal de Santa Catarina, Santa Catarina, 2001

[9] Stephen A. Thomas, "*IPng and the TCP/IP Protocols - Implementing the Next Generation Internet",* Wiley Computer Publishing, 1996

[10] Ferguson, Paul., Huston, Geoff (1998) *Quality of Service. Delivering QoS on the Internet and in Corporate Networks.* John Wiley Computer Publishing.

[11] Osborne, E., Engenharia de Tráfego com MPLS, Cisco Press, 2003

[12] ISO/IEC DIS 13236, Information Technology - Quality of Service – Framework, ISO/OSI/ODP, Julho 1995.

[13] Vogel, L. A. et al., *Distributed Multimedia and QoS: A Survey, IEEE Multimedia,* Verão 1995.

[14] Teitelman, B. & Hanss, T., QoS Requirements for Internet2, Internet2 QoS Work Group Draft, Abril 1998.

[15] Stardust.com, Inc., *The Need for QoS*, White Paper, Julho 1999.

[16] Cisco, *Internetworking Technologies Handbook*

[17] Santana,G. e Dantas,M., *Hybrid Quality of Service Architectures for IP Networks,*Universidade de Brasilia, 2001.

[18] Kamienski, Carlos Alberto & Sadok, Djamel , Qualidade de Serviço na Internet, SBRC 2000, Belo Horizonte, Maio 2000.

[19] H. Zhang. *Service Disciplines For Guaranteed Performance Service in Packet-Switching Networks ,Proceedings of the* IEEE, 83(10), Oct 1995

[20] L. Zhang, S. E. Deering, D. Estrin, S. Shenker and D. Zappala. RSVP: A New Resource *ReSerVation Protocol*. IEEE Network Magazine, vol. 9 nº 5, September, 1993

[21] IETF "*Differentiated Services Working Group"*, Em http://www.ietf.org/html.charters/ e http://www.ietf.org/ids.by.wg/

[22] Nichols, K et all (1998) *Definition of the Differentiated Services Field (DS Field) in the IPv4 and IPv6 Headers*. RFC 2474. Dezembro de 1998. http://www.ietf.org/rfc/rfc2474.txt.

[23] Zhang,L., et al., RFC 2205 - Resource *ReSerVation Protocol* (RSVP) -Version 1, Functional Specification

[24] IETF "*Multiprotocol Label Switching Working Group*". Em http://www.ietf.org/html.charters/ e http://www.ietf.org/ids.by.wg/

[25] Andersson,L; Doolan,P.; Feldman,N.; Fredette, A.;Thomas, B., , : *LDP Specification*. RFC 3036, Jnaeiro 2001

[26] Shenker, S., Wroclawski, J., General Characterization Parameters for Integrated Service Network Elements. RFC-2215, Setember 1997.

[27] Apostolopoulos, G. et. al., *Quality of Service Based Routing: A Performance Perspective,* ACM SIGCOMM '98, Agosto, 1998.

[28] Wroclawski, J., *Specification of the Controlled-Load Network Element Service,* RFC 2211, Setembro 1997.

[29] Black, U., *Multi-Protocol Label Switching*. Prentice Hall, December 2000.

[30] Rosen, E., Viswanathan, A. and R. Callon, "Multiprotocol Label Switching Architecture", RFC 3031, January 2001.

[31] Magalhães, M., Cardozo, E., Comutacão IP por Rótulos através de MPLS (Minicurso). In Anais do 19º Simpósio Brasileiro de Redes de Computadores, 2001.

[32] Stardust Technologies, Inc. (1999): *QoS protocols & architectures*, White Paper.

[33] Oliveira, Sandro Silva de. "Análise de tráfego na integração de redes IP e ATM usando simulação". *Dissertação de Mestrado*. Programa de Pós-Graduação em Ciência da Computação/UFSC, Out. 2001

[34] Topke, Claus Rugani, "Uma metodologia para caracterização de Tráfego e Medidas de Desempenho em *Backbones IP"*. Dissertação de Mestrado. COPPE / UFRJ, Rio de Janeiro, 2001

[35] Implementação de Serviços Diferenciados em uma Rede Local César Augusto de Oliveira Soares1 Rosivelt Alves do Carmo1

[36] Heinanen, J., Baker, F., Weiss, W., Wroclawski, J., *"Assured Forwarding PHB Group*", RFC 2597

[37] Cisco Systems, Inc., *Cisco IOS Quality of Service Solutions Configuration Guide*, Release 12.2, 2001

[38] Allman, M., Paxson, V. & Stevens W., "*TCP Congestion Control"*, RFC 2581, Abril 1999.

[39] Paxson, V., et. al., "*Known TCP Implementation Problems*", Internet RFC 2525, Março 1999.

[40] Braden, R., et. al, "*Recommendations on Queue Management and Congestion Avoidance in the Internet"*, RFC 2309, Abril 1998.

[41] Jacobson, V., *"Congestion avoidance and control*", ACM SIGCOMM'88, Agosto 1988.

[42] Jacobson, V., Nichols, K. & Poduri, K., "*An Expedited Forwarding*

PHB", RFC 2598, Junho 1999.

[43] Jain, R., "*Myths about Congestion Management in High Speed Networks*", Internetworking: Res. And Exp., vol.3, 1992.

[44] Lefelhocz, C. et. al., *Congestion Control for Best-Effort Service: Why We Need A New Paradigm*, *IEEE Network*, Janeiro 1996.

[45] Thomson, K., Miller, G. J. & Wilder, R., *"Wide-Area Internet Traffic Patterns and Characteristics, IEEE Network*", Novembro 1997.

[46] Yang, C.-Q. & Reddy, A. V. S., "A Taxonomy for Congestion Control Algorithms in Packet Switching Networks", IEEE Network Magazine, Julho 1995.

[47] Ferguson, P. & Huston, G. (1999): *Quality of Service: Delivering QoS on the Internet and in Corporate Networks* - Wiley Computer Publishing.

[48] Linney, L. (1999): Differentiated Services on IBM 221x Routers, IBM Co.

[49] Ardeola, F. R & Tarouco, L. M.," Priorização de Tráfego em Redes IP", UFRJ, 1999

[59] Cisco System Inc., *Internetworking Technologies Handbook*, 2003. Documento disponível em http://www.cisco.com

[50] Cisco System Inc., Classification Overview, release 12.1, 2002. Documento disponível em http://www.cisco.com

[51] Cisco System Inc., Policing and Shaping Overview, release 12.1, 2002. Documento disponível em http://www.cisco.com

[52] Cisco System Inc., Quality of Service Overview, release 12.1, 2002. Documento disponível em http://www.cisco.com

[53] Parada,C., Fontes, F., Carapinha, J., Lima, S., Carvalho, P., Comparação de Plataformas para Suporte de Serviços Diferenciados em Redes IP, Conferência de Redes de Computadores (CRC2001), 2001

[54] Nichols, K., Carpenter, B., *RFC 3086 – Definition of the Differentiated Services PerDomain Behaviors and Rules for their Specification*, Abril 2001.

[55] Ferrari, T.,Chimento, P., *A Measurement-based Analysis of Expedited Forwarding PHB Mechanisms*, TERENA 2000.

[56] Ferrari, T., Paul, G., Raffaelli, C., *PriorityQueuing Applied to Expedited Forwarding: aMeasurement-based Analysis*, 2000.

[57] Cisco Systems (Documentation), *Cisco IOS (TM) Software - QoS Solutions*, 2000.

[58] Floyd, S., Jacobson, V., *Random Early Detection gateways for Congestion Avoidance*,IEEE/ACM Transactions on Networking, V.1 N.4, Agosto 1993, pp 397-413.

[59] Makkar, R., Lambadaris, I., J. Salim, N. Seddigh, B. Nandy, J. Babiarz, *Empirical Study of Buffer Management Scheme for DiffServ Assured Forwarding PHB*, ICCCN2000.

[60] Naval Research Laboratory (NRL). *The Multi-Generator (MGEN) Toolset,* Disponível em http://manimac.itd.nrl.navy.mil/MGEN.

[61] Cisco System Inc., Low Latency Queueing, 1999. Documento disponível em http://www.cisco.com

[62] Hewlett-Packard Company, (1995**).** *Netperf: A Network Performance Benchmark, D*isponível em *http://www.netperf.org*

[63] Networks Associates Technology, Inc (NAI), Sniffer Portable 4.7

[64] Adailton J. S. Silva, Qualidade de Serviço em VOIP – Parte 2, NewsGeneration, 27/09/2000, vol 4, numero 5.

[65] Garcia, A., Sudre, G., Gerenciamento de Redes Baseado em SLA´s e Procedimento Remoto: Uma Nova Realidade, SUCESU-ES, 25/06/2004

[66] Pinheiro, J.M.Santos, "Afinal, o que é Qualidade de Serviço?". www.projetoderedes.com.br, março 2004

[67] Oliveira, R.D, Farines, J.M, Medições e Testes para Aplicações Envolvendo Mídias Contínuas em Redes IP com Serviços Diferenciados, XXII Congresso da Sociedade Brasileira de Computação, florianópolis 2002

[68] Wroclawski, J., *Specification of Guaranteed Quality of Service,* RFC 2212, Setembro 1997

[69] Wroclawski, J., *Integrated Services Management Information Base using SMIv2,* RFC 2213, Setembro 1997

[70] Postel, J., User Datagram Protocol, RFC 768, USC/Information Sciences Institute, Agosto 1980

[71] Postel, J., User Datagram Protocol, RFC 793, USC/Information Sciences Institute, Setembro 1981

[72] Postel, J., The TCP Maximum Segment Size and Related Topics, RFC 869, USC/Information Sciences Institute, Novembro 1983

[73] Caporal, C., Introdução à Redes de Computadores Introdução à Família de Protocolos TCP/IP Implementação de Serviços de Alto Nível Gerência e Segurança de Redes, Santa Catarina-Brasil

[74] Y. Bernet, P. Ford, R. Yavatkar, F. Baker, L. Zhang, M. Speer, R. Braden, B. Davie, J. Wroclawski, E. Felstaine, *A Framework for Integrated Services Operation over Diffserv Networks,* RFC 2998, November 2000.