DANIEL CARDOSO GOMES

PROPOSTA DE OTIMIZAÇÃO DO TRÁFEGO DA REDE DA UNIVERSIDADE FEDERAL DE LAVRAS UTILIZANDO A TÉCNICA DE SPANNING TREE PROTOCOL

Monografia de graduação apresentada ao Departamento de Ciência da Computação da Universidade Federal de Lavras como parte das exigências do curso de Ciência da Computação para obtenção do título de Bacharel em Ciência da Computação.

LAVRAS
MINAS GERAIS – BRASIL
2005

DANIEL CARDOSO GOMES

PROPOSTA DE OTIMIZAÇÃO DO TRÁFEGO DA REDE DA UNIVERSIDADE FEDERAL DE LAVRAS UTILIZANDO A TÉCNICA DE SPANNING TREE PROTOCOL

Monografia de graduação apresentada ao Departamento de Ciência da Computação da Universidade Federal de Lavras como parte das exigências do curso de Ciência da Computação para obtenção do título de Bacharel em Ciência da Computação.

Área de Concentração:
Rede de Computadores

Orientador:
Prof. Rêmulo Maia Alves

Co-orientador:
Prof. Anderson Bernardo dos Santos

LAVRAS
MINAS GERAIS – BRASIL
2005

Ficha Catalográfica preparada pela Divisão de Processos Técnicos da Biblioteca Central da UFLA

Gomes, Daniel Cardoso

Proposta de Otimização do Tráfego da Rede da Universidade Federal de Lavras utilizando Spanning Tree Protocol. Lavras – Minas Gerais, 2005. 103 páginas.

Monografia de Graduação – Universidade Federal de Lavras. Departamento de Ciência da Computação.

1. Introdução. 2. Referencial Teórico. 3. Materiais e Métodos. 4. Resultados e Discussões. 5. Conclusão. I. GOMES, D. C. II. Universidade Federal de Lavras.

DANIEL CARDOSO GOMES

PROPOSTA DE OTIMIZAÇÃO DO TRÁFEGO DA REDE DA UNIVERSIDADE FEDERAL DE LAVRAS UTILIZANDO SPANNING TREE PROTOCOL

Monografia de graduação apresentada ao Departamento de Ciência da Computação da Universidade Federal de Lavras como parte das exigências do curso de Ciência da Computação para obtenção do título de Bacharel em Ciência da Computação.

Aprovada em 30 de Junho de 2005.

_____
Prof. Anderson Bernardo dos Santos

_____
Rafael de Magalhães Dias Frinhani

_____
Prof. DSc.Rêmulo Maia Alves (Orientador)

LAVRAS
MINAS GERAIS – BRASIL
2005

AGRADECIMENTOS

Aos meus pais, por todo apoio e dedicação que recebi durante toda vida acadêmica.

Aos professores, que participaram ativamente e contribuíram para que este projeto fosse concluído.

À minha Lindinha, por toda força e incentivo e principalmente por estar junto em todos os momentos importantes dessa luta.

Muito obrigado!

RESUMO

Proposta de Otimização do Tráfego da Rede da Universidade Federal de Lavras utilizando a técnica de Spanning Tree Protocol

Com a expansão das organizações e de suas redes computacionais, sua alta disponibilidade é um requisito chave. Até mesmo curtos períodos de inatividade de uma rede podem gerar perdas de produtividade. Diante disso, o presente trabalho apresenta uma proposta de otimização do tráfego da Rede da Universidade Federal de Lavras, através da implantação de redundâncias e a posterior implantação da técnica de *Spanning Tree Protocol* (STP). Os testes realizados em laboratório contribuíram para avaliar a viabilidade da utilização em campo desta tecnologia e mostraram que o tempo gasto pelo STP para restabelecer os serviços de rede é insignificante se comparado ao tempo que geralmente a Rede Ufla fica indisponível após a ocorrência de uma falha.

Palavras-Chave: Rede de Computadores, *Spanning Tree Protocol*, Otimização, Redundância.

ABSTRACT

Optimization proposal of the Federal University of Lavras Network's traffic using the Spanning Tree Protocol technique

With the expansion of the organizations and of their computer networks, their high availability is a key requirement. Even short periods of inactivity of a network can generate productivity losses. Before that, the present work presents a optimization proposal of the Federal University of Lavras network's traffic, through the implantation of redundancies and the subsequent implementation of the Spanning Tree Protocol (STP) technique. The tests accomplished at laboratory contributed to evaluate the viability of the use in field of this technology and they showed that the time spent for STP to reestablish the network services is insignificant if compared at the time that usually the Ufla's Network is unavailable after the occurrence of a flaw.

Keywords*: Computer Networks, Spanning Tree Protocol, Optimization, Redundancy.*

SUMÁRIO

LISTA DE FIGURAS

LISTA DE TABELAS

1 INTRODUÇÃO

1.1 Visão Geral

Em tempos em que a competitividade faz com que as organizações preocupem-se cada vez mais com a racionalização e o aproveitamento máximo de seus recursos, a fim de obter ganhos de eficiência, é imprescindível a procura constante de novas soluções. E com o aumento da complexidade das redes de computadores, surgiu a necessidade de realizar um gerenciamento de redes mais eficiente e abrangente, visando manter a disponibilidade e consistência dos serviços baseados em redes de computadores.

Dentro desta visão, em que o ganho de performance de uma rede é perseguido constantemente, pode-se lançar mão de recursos já disponíveis no mercado. Quando projetos de redes utilizam múltiplos *switches*, a maioria dos engenheiros de redes inclui segmentos redundantes entre os *switches*. O objetivo disso é simples: criar enlaces alternativos para o tráfego de dados. Um *switch* pode falhar, um cabo pode ser cortado ou desconectado, e se houver um enlace redundante na rede, o serviço ainda estará disponível para a maioria dos usuários.

Porém, projetos de redes com enlaces físicos redundantes podem fazer com que os dados nessa rede entrem em *loop* infinito, ou seja, permaneçam em tráfego constante, congestionando a rede e gerando problemas significativos de performance.

Para minimizar esses problemas pode-se implementar uma técnica de otimização de tráfego em redes chamada *Spanning Tree Protocol* (STP) para impedir que os dados trafeguem indefinidamente pelos enlaces redundantes. Este recurso acompanha a grande maioria dos *switches*, porém, é ignorado e até desprezado como recurso de aumento de performance pelo público comprador.

De acordo com esta visão, buscou-se analisar a situação da rede da Universidade Federal de Lavras (Ufla), uma instituição pública que, ao contrário de muitas outras, procura fazer o melhor uso possível dos equipamentos e recursos que possui, aproveitando o fato de ser uma organização educacional e possuir centenas de pesquisadores, muitos deles da área de Tecnologia de Transporte de Informações.

Estudando a configuração da rede na Ufla, observou-se que esta não possui uma boa topologia e nenhum recurso de gerenciamento de tráfego implementado. Isso ocasiona

diversos problemas, como tráfego intenso e sem controle, interrupção de serviços, sobrecarga e lentidão, entre outros.

Uma boa performance de rede depende de vários fatores, e entre eles está a customização dos equipamentos ativos, visando personalização e ganho de funcionalidades. O desafio é considerar todas as possibilidades e parâmetros, como custo, segurança, gerenciamento, entre outros. Este desafio torna-se ainda maior quando se trata de uma Rede Campus de uma Universidade Pública, onde os recursos para melhorias são limitados.

Diante disso, o presente trabalho elaborou uma proposta de otimização de tráfego de dados na rede Ufla, através da implementação da técnica de *Spanning Tree Protocol* utilizando equipamentos já disponíveis – os *switches*.

Foi feita uma análise minuciosa da Rede Ufla, identificando os principais problemas enfrentados tanto pelos administradores quanto pelos usuários, as adaptações necessárias para possibilitar a implementação do STP e os pontos de maior necessidade de melhoria. Um teste-piloto foi realizado em laboratório para validar os benefícios e a funcionalidade da técnica.

Os resultados obtidos contribuíram consideravelmente não só para a viabilidade do aumento de performance da rede implementando a técnica testada, mas para conhecer e documentar as características estruturais e peculiares da Rede Ufla.

Outro ponto que merece destaque é que o maior desafio de se conseguir a implementação do STP na Ufla é fazer as novas conexões entre os equipamentos, de forma a criar redundâncias na rede.

Apesar disso, o objetivo deste trabalho foi alcançado. Os resultados do teste-piloto forneceram os subsídios necessários para a implantação do projeto na Rede Ufla e a documentação necessária para otimizar a sua performance.

1.2 Objetivo

O objetivo deste trabalho foi propor uma solução para aumentar o desempenho da Rede da Universidade Federal de Lavras através da implementação da técnica de *Spanning Tree Protocol* em *switches* Ethernet, sobre uma topologia redundante.

1.3 Escopo do Trabalho

O Capítulo 2 traz todo o referencial teórico necessário ao entendimento do desenvolvimento deste trabalho.

O Capítulo 3 descreve os materiais e métodos utilizados, os passos necessários à execução deste projeto e todo o seu processo de desenvolvimento.

O Capítulo 4 mostra os resultados alcançados e algumas discussões sobre os mesmos.

O Capítulo 5 apresenta a conclusão do trabalho, ou seja, uma visão geral de todo este projeto, descrevendo os principais pontos discutidos ao longo do mesmo.

2 REFERENCIAL TEÓRICO

2.1 Redes de Computadores

Hoje em dia, tanto no ambiente explícito da informática quanto fora dele, todos nós temos contato com algum tipo de rede em maior ou menor grau. As redes de computadores surgiram da necessidade da troca de informações, onde é possível ter acesso a um dado que está fisicamente localizado longe de você (TORRES, 2001).

"A network consists of computers, called nodes or stations. The computers are connected to, or can communicate with, each other in some way. Nodes run special software for initiating and managing network interactions. With the help of networking software, nodes can share files and resources" (FEIBEL, 1996).

Segundo SOARES (1995), uma rede de computadores é formada por um conjunto de módulos processadores e por um sistema de comunicação, ou seja, é um conjunto de enlaces físicos e lógicos entre vários computadores (chamados *hosts*).

Além da vantagem de se trocar dados, há também a vantagem de compartilhamento de periféricos, que podem significar uma redução nos custos de equipamentos. A Figura 2.1 representa um exemplo de compartilhamento de impressora (periférico) que está sendo usado por vários computadores.

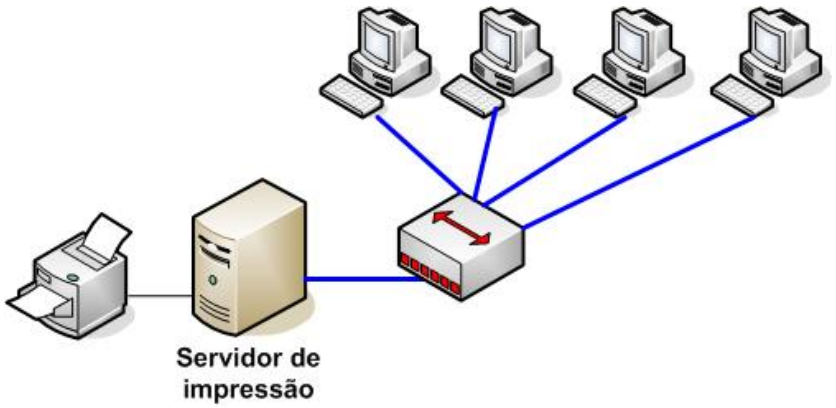


Figura 2.1: Exemplo de compartilhamento de periféricos.

TANENBAUM (1997) enfatiza que a facilidade de se trocar dados e compartilhar periféricos, como impressoras ou scanners, é o motivo básico de uma rede, podendo significar uma redução nos custos dos equipamentos e aumentar a confiabilidade do sistema, pois tem fontes alternativas de fornecimento.

As redes de computadores permitem que as aplicações distribuídas utilizem *login* remoto, correio eletrônico, transmissão de áudio e vídeo em tempo real (como em uma videoconferência), jogos distribuídos, a *World Wide Web* e muito mais (KUROSE & ROSS, 2003).

Segundo SOARES et. al. (1995), o sistema de comunicação em uma rede constitui-se de um arranjo topológico interligando os vários módulos processadores através de enlaces físicos (meios de comunicação) e de um conjunto de regras com o fim de organizar a comunicação (protocolos).

A possibilidade de mesclar informações, comunicação e entretenimento certamente dará origem a uma nova e avançada indústria baseada nas redes de computadores (TANENBAUM, 1997).

As redes de computadores podem ser classificadas em: LAN, MAN e WAN. Cada uma dessas classificações será explicada a seguir.

2.1.1 Local Area Network – LANs

As redes locais, também chamadas de LANs, são redes privadas contidas em um prédio ou em um campus universitário que tem alguns quilômetros de extensão. Elas são amplamente usadas para conectar computadores pessoais e estações de trabalho em escritórios e instalações industriais (TANENBAUM, 1997).

Segundo TANENBAUM (1997), a tecnologia de transmissão das LANs quase sempre consiste em um cabo ao qual todas as máquinas são conectadas, como acontece com as extensões telefônicas que já foram usadas nas áreas rurais. As velocidades de transmissão em redes locais mais comuns são 10 Mbps, 100 Mbps e 1Gbps (Gigabits por segundo).

A Figura 2.2 mostra um esquema de uma LAN.

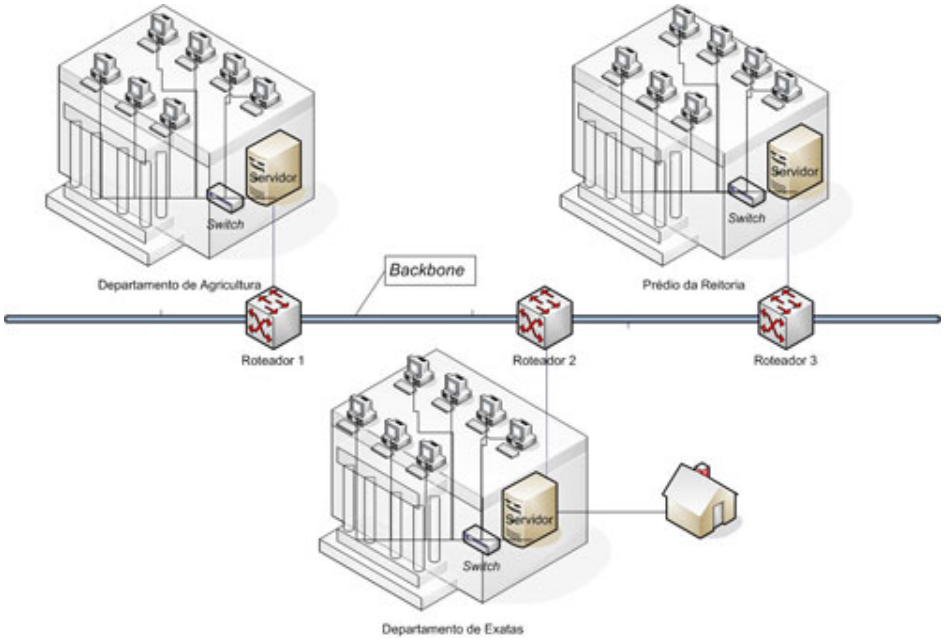


Figura 2.2: Esquema de uma rede local.

2.1.2 Metropolitan Area Network – MANs

Metropolitan Area Network ou Redes Metropolitanas são redes de alta velocidade que podem transportar voz, dados e imagens a uma velocidade de até 200 Mbps (Megabits por segundo) ou ainda maior em distâncias de até 75 km. Essa velocidade poderá variar de acordo com a arquitetura da rede.

As MANs podem ser utilizadas para interligar dois ou mais pontos dentro de uma mesma cidade. Essas redes representam o principal meio de se ter, num curto espaço de tempo, um *backbone*[1] nacional de alta velocidade estabelecido no país.

TANENBAUM (1997) explica que uma rede metropolitana, ou MAN, é, na verdade, uma versão ampliada de uma LAN, pois basicamente os dois tipos de rede utilizam tecnologias semelhantes. Uma MAN pode abranger um grupo de escritórios vizinhos ou uma cidade inteira e pode ser privada ou pública. Esse tipo de rede é capaz de transportar dados e voz, podendo inclusive ser associado à rede de televisão a cabo local.

As redes metropolitanas não podem fugir do conceito de uma grande rede local, na qual todos os serviços que estariam disponíveis sobre seu próprio *Switch*[2] local se tornam

[1] Um backbone é a "espinha dorsal" das grandes redes de comunicação na Internet. Normalmente são conexões de alta-velocidade que interconectam redes regionais.

disponíveis também em longas distâncias. Esta opção de rede também é uma grande aliada das empresas que desejam replicar seus sites por questões de conectividade ou segurança.

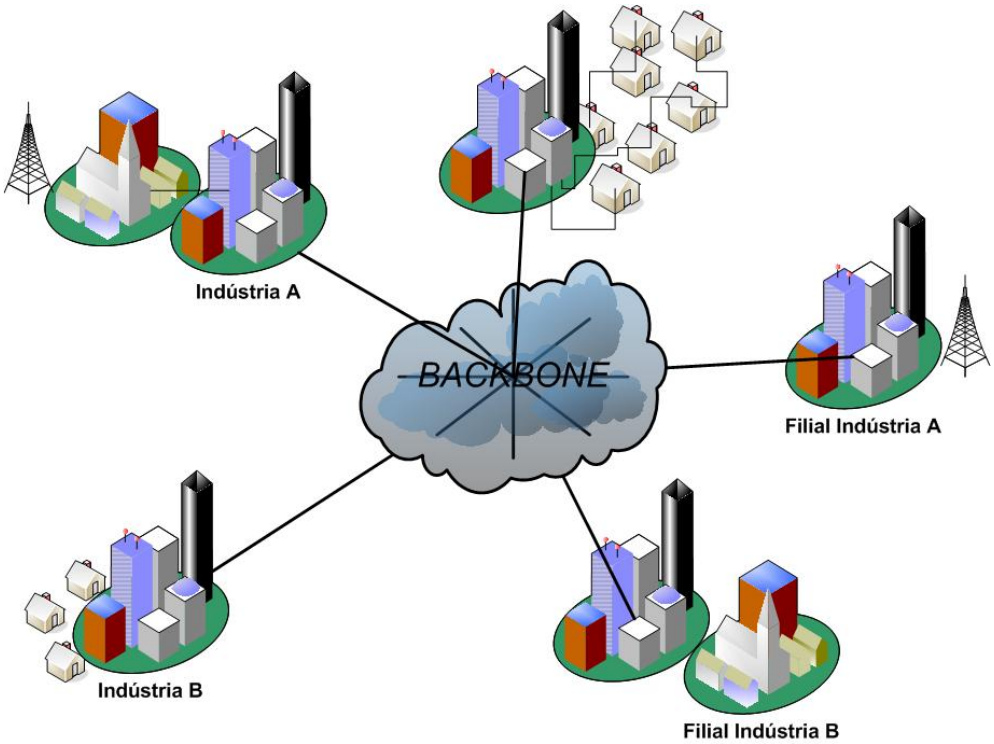


Figura 2.3: Esquema de uma rede MAN.

2.1.3 Wide Area Network – WANs

Uma rede geograficamente distribuída, ou WAN, abrange uma ampla área geográfica (um país ou continente). Ela contém um conjunto de máquinas cuja finalidade é executar as aplicações do usuário (TANENBAUM, 1997). A Figura 2.4 ilustra um esquema de uma WAN.

[2] Equipamento de interconexão que permite comunicação múltipla entre todos as máquinas conectadas a ele.

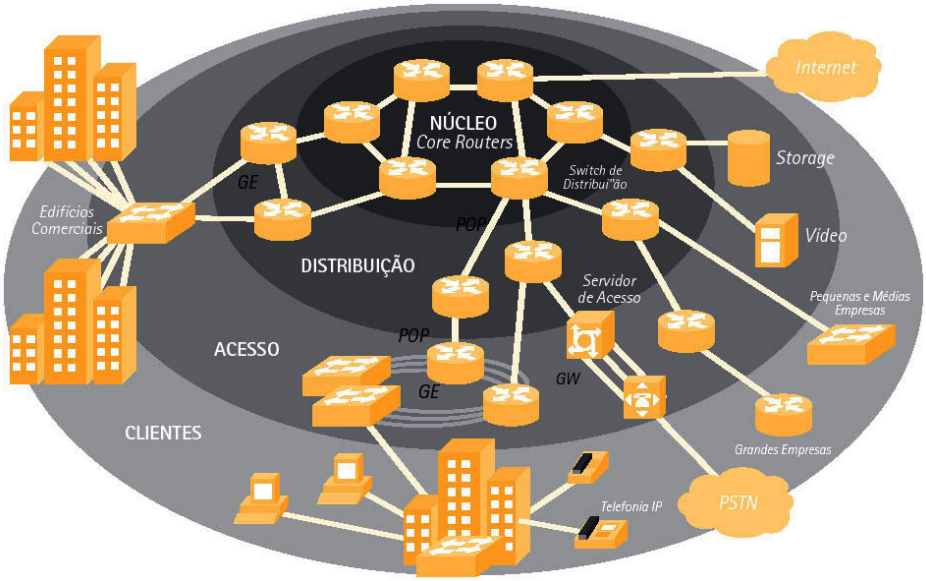


Figura 2.4: Modelo conceitual de uma rede WAN.

Fonte: *Business & Technology Review* Ano 03 - Nº 06.

2.2 Topologias de Rede

Segundo SOARES et. al. (1995), a topologia de uma rede refere-se à forma como os enlaces físicos e os nós de comutação estão organizados, determinando os caminhos físicos existentes e utilizáveis entre quaisquer pares de estações conectadas a essa rede.

A topologia de uma rede descreve como o é o *layout* do meio através do qual há o tráfego de informações, e também como os dispositivos estão conectados a ele. São várias as topologias existentes, podemos citar a Topologia em Barramento, Estrela, Anel, Malha, e Topologias Híbridas.

2.2.1 Topologia em Anel

Nessa topologia, procura-se diminuir ao máximo o número de ligações no sistema. As estações são ligadas ponto a ponto e operam em um único sentido de transmissão, como pode ser visto na Figura 2.5. Uma mensagem deverá circular pelo anel até que chegue ao módulo de destino, sendo passada de estação em estação (SOARES et. al., 1995).

Tal topologia apresenta limitações de velocidade e confiabilidade. Caso uma rede distribuída aumente consideravelmente o número de estações, isso significa um aumento intolerável no tempo de transmissão. Outro fator limitante refere-se à inexistência de

caminhos alternativos para o tráfego de informações. Se porventura um segmento do anel for cortado, toda a rede fica comprometida.

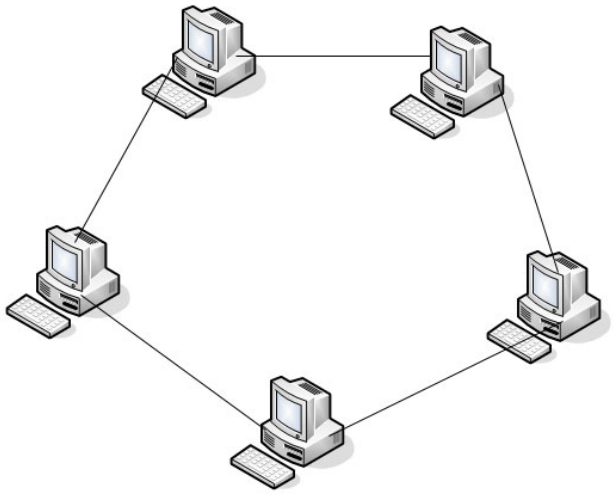


Figura 2.5: Topologia em anel.

2.2.2 Topologia em Malha

Nesta topologia todos os nós estão conectados a todos os outros nós, como se estivessem entrelaçados (Figura 2.6). Já que são vários os caminhos possíveis por onde a informação pode fluir da origem até o destino, este tipo de rede está menos sujeita a erros de transmissão, o tempo de espera é reduzido, e eventuais problemas não interrompem o funcionamento da rede.

Um problema encontrado é com relação às interfaces de rede, já que para cada segmento de rede seria necessário instalar, numa mesma estação, um número equivalente de placas de rede. Como este tipo de topologia traz uma série de desvantagens para a maioria das instalações, raramente é usado.

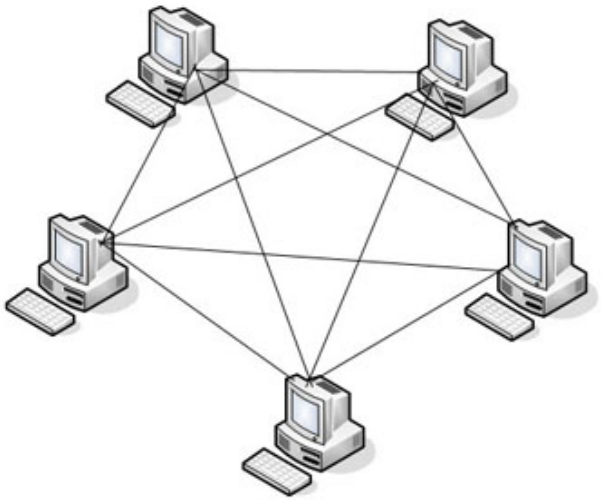


Figura 2.6: Topologia em Malha.

2.2.3 Topologia em Barramento

Na topologia em barramento, todas as estações compartilham um mesmo cabo. A barra é geralmente compartilhada em tempo e freqüência, permitindo transmissão de informação.

Esta topologia é caracterizada por uma linha única de dados (o fluxo é serial), finalizada por dois terminadores (casamento de impedância), na qual cada nó é conectado de tal forma que toda mensagem enviada passa por todas as estações, sendo reconhecida somente por aquela que está cumprindo o papel de destinatário (estação endereçada).

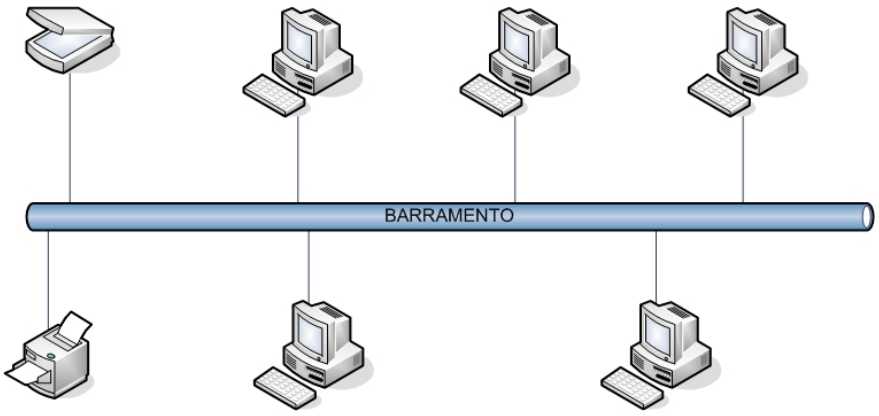


Figura 2.7: Topologia em Barramento.

O desempenho de um sistema que usa topologia barramento é determinado pelo meio de transmissão, número de estações conectadas, controle de acesso, tipo de tráfego, entre outros (SOARES et. al., 1995).

2.2.4 Topologia em Estrela

A topologia estrela, mostrada na Figura 2.8, é caracterizada por um elemento central que gerencia o fluxo de dados da rede, estando diretamente conectado (ponto-a-ponto) a cada nó, daí surgiu a designação "Estrela". Toda informação enviada de um nó para outro deverá obrigatoriamente passar pelo ponto central, ou concentrador, tornando o processo muito mais eficaz, já que os dados não irão passar por todas as estações. O concentrador encarrega-se de rotear o sinal para as estações solicitadas, economizando tempo.

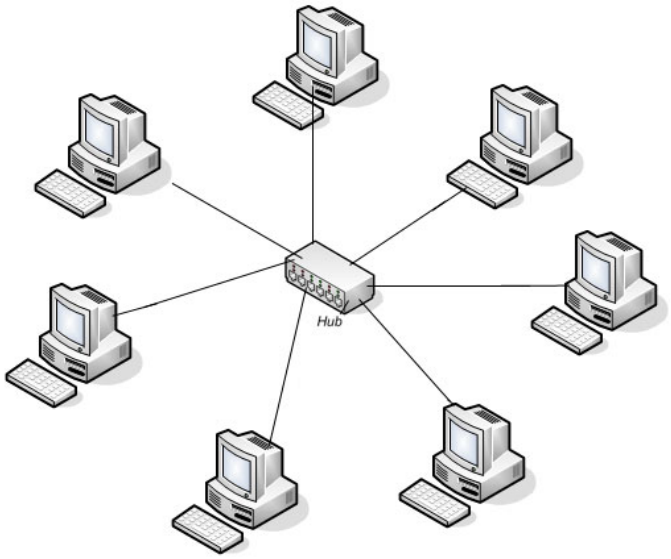


Figura 2.8: Topologia em Estrela.

Redes em estrela podem atuar por difusão (*broadcasting*) ou não. Em redes de difusão, todas as informações são enviadas ao nó central, que é responsável por distribui-las a todos os nós da rede (SOARES et. al., 1995).

Uma vez que o sinal sempre será conduzido para um elemento central, e a partir deste para o seu destino, as informações trafegam rapidamente, sendo assim, as mais indicadas para redes em que imperam o uso de informações "pesadas", como a troca de

registros de uma grande base de dados compartilhada, som, gráficos de alta resolução e vídeo.

Segundo SOARES et. al. (1995), as vantagens oferecidas na prática são muitas: a instalação de novos segmentos não requer muito trabalho, a localização de problemas fica mais fácil; a rede estrela é mais fácil de dispor fisicamente mediante as dificuldades encontradas no ambiente de trabalho (no momento de instalação, expansão, e mesmo se a rede tiver de ser deslocada); se um problema ocorrer num segmento os outros permaneceram em atividade; e, como já foi dito, a rede estrela geralmente oferece taxas de transmissão maiores. Toda rede cliente-servidor, como pode ser notado, segue a topologia estrela.

2.3 Protocolo TCP/IP

Um protocolo define o formato e a ordem das mensagens trocadas entre duas ou mais entidades comunicantes, bem como as ações realizadas na transmissão e/ou no recebimento de uma mensagem ou outro evento (KUROSE & ROSS, 2003). Em outras palavras, protocolo é um conjunto de regras que determinam como será feita a comunicação entre os computadores em uma rede.

O protocolo TCP/IP (*Transmission Control Protocol/Internet Protocol*) atualmente é o mais usado em redes locais. Isso se deve basicamente à popularização da internet já que ele foi criado para este fim.

Uma das grandes vantagens do TCP/IP em relação a outros protocolos existentes, segundo TORRES (2001), é que ele é roteável, possibilitando o envio de mensagens ou dados em redes grandes de longa distância, onde pode haver vários caminhos para atingir o computador receptor.

Outro fato que tornou o TCP/IP popular é que ele possui arquitetura aberta e qualquer fabricante pode adotar sua própria versão de TCP/IP em seu sistema operacional, como afirma Richard Stevens, em STEVENS (1993): “O TCP/IP é na verdade um conjunto de protocolos que permite que computadores de todos os tamanhos, de diferentes fabricantes rodando sistemas operacionais totalmente diferentes, comuniquem-se entre si ”.

As principais características do TCP/IP, segundo STEVENS (1993), são:

- Padrão aberto, livremente disponível e desenvolvido independentemente do hardware ou sistema operacional do computador;
- Independente de hardware específico de rede;

- Possui um esquema de endereçamento comum que possibilita todo dispositivo comunicar-se a outro em toda rede, mesmo que esta seja tão grande quanto a Internet;
- É composto por protocolos padronizados de alto nível para serviços consistentes e amplamente disponíveis.

TANENBAUM (1997) nos leva a compreender que, para reduzir a dificuldade de projeto, os projetistas de rede organizam os protocolos – e o hardware e o software de rede que implementam os protocolos – em camadas. Com uma arquitetura de protocolo em camadas, cada protocolo pertence a uma das camadas. É importante entender que um protocolo de uma camada n é distribuído entre as entidades da rede que implementam aquele protocolo. Duas arquiteturas merecem destaque neste trabalho: o OSI (*Open System Interconnection),* um modelo teórico, elaborado pela ISO (*International Organization for Standardization*) e o TCP/IP, mais prático, que é especificado como um sistema de 4 camadas. A Figura 2.9 mostra as camadas de cada modelo:

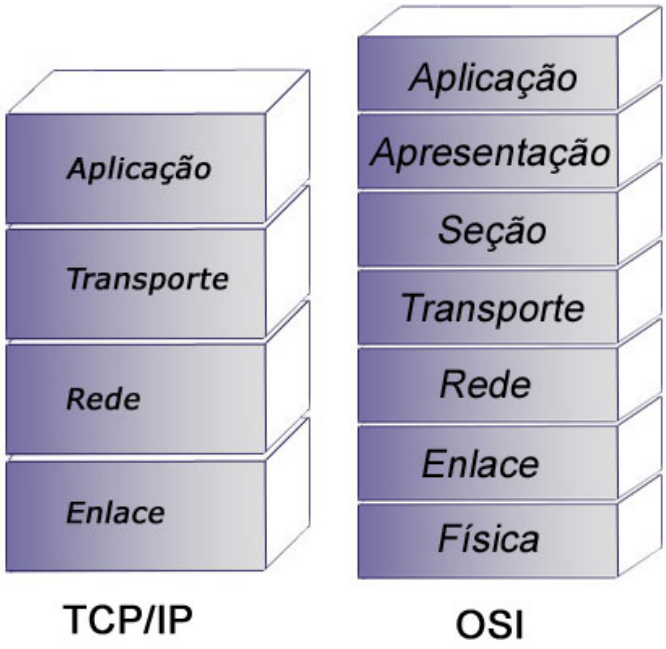


Figura 2.9: Camadas dos modelos TCP/IP e OSI.

Este trabalho detalha somente as camadas e características intrínsecas ao modelo TCP/IP, pelo fato de ser o mais utilizado na prática. Suas camadas são discutidas a seguir.

2.3.1 Camada de Enlace (*Link Layer*)

A camada de Enlace, também chamada de camada de Interface com a rede ou camada de abstração de hardware. Tem como principal função a interface do modelo TCP/IP com os diversos tipos de redes (ATM, FDDI, Ethernet, Token Ring). Inclui, normalmente, o *driver* de dispositivo no sistema operacional e a sua interface de rede correspondente no computador. Juntos eles tratam todos os detalhes de hardware, e a comunicação física com a mídia de transmissão utilizada (STEVENS, 1993).

É responsável por enviar o datagrama recebido da camada de Rede em forma de quadro (*frame*) através da rede, como mostra a Figura 2.10.

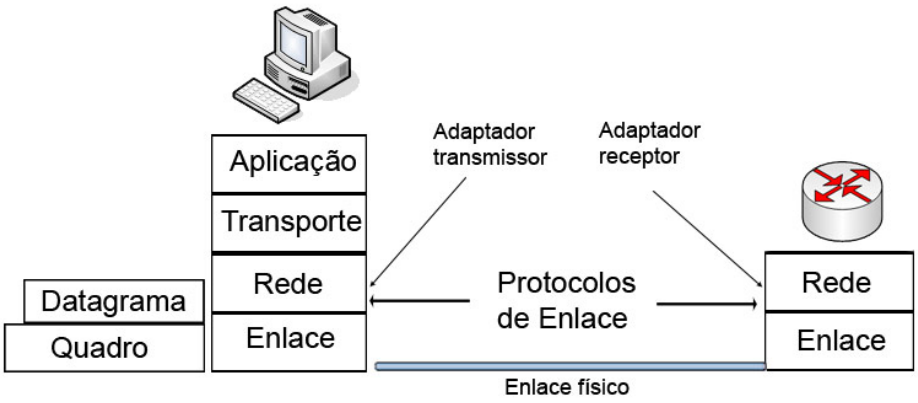


Figura 2.10: Esquema da função da camada de enlace.

2.3.2 Camada de Rede (*Network Layer*)

Essa camada integra toda a arquitetura da rede. Sua tarefa é permitir que as estações enviem pacotes em qualquer rede e garantir que eles sejam transmitidos independentemente do destino (que pode ser outra rede).

A camada de rede, também conhecida por camada Inter-Redes, define um formato de pacote oficial – o datagrama – e um protocolo chamado IP (*Internet Protocol*). Entregar pacotes IP onde eles forem requeridos é a tarefa da camada de rede.

Também é de responsabilidade desta camada o roteamento de pacotes, isto é, adicionar ao datagrama informações sobre o caminho que ele deverá percorrer.

Além do protocolo IP, vários outros operam nessa camada: ICMP (*Internet Control Message Protocol*), ARP (*Address Resolution Protocol*) e RARP (*Reverse Address Resolution Protocol*).

ARP (*Address Resolution Protocol*)

Para que sistemas IP possam se comunicar entre si, eles primeiramente precisam ser capazes de identificar os endereços MAC[3] dos outros dispositivos localizados no mesmo segmento de rede. Neste sentido, é necessário um protocolo que possa associar um endereço proprietário de uma interface, ao seu endereço IP correspondente. Este serviço é realizado pelo protocolo de resolução de endereços ou ARP.

O protocolo ARP é responsável por fazer a conversão entre os endereços IPs e os endereços MAC da rede, como explica TORRES (2001), as redes baseiam-se em um endereçamento virtual – o endereçamento IP. Em contrapartida, as placas de rede das máquinas conectadas à rede usam o esquema de endereçamento MAC.

A ação do protocolo ARP é detectar o endereço da placa de rede para a qual o pacote deve ser entre, já que no pacote há somente o endereço IP de destino e não o endereço MAC da placa de rede.

O ARP funciona mandando primeiramente uma mensagem de broadcast para toda a rede perguntando, a todas as maquinas, qual responde pelo endereço IP para o qual se pretende transmitir um pacote. Então, a máquina que corresponde a tal endereço responde, identificando-se e informando seu endereço MAC para que a transmissão de dados possa ser estabelecida.

O módulo ARP funciona recuperando endereços de camada de enlace (IP) a partir dos endereços de camada física (MAC), pesquisando em toda rede para determinar o endereço correspondente que está sendo convertido (KUROSE & ROSS, 2003).

IP (*Internet Protocol*)

Um papel essencial desse protocolo é receber os dados enviados pela camada de transporte e enviá-los para a camada de enlace. No módulo IP, os dados são empacotados em datagramas, que ao chegarem na camada de enlace serão empacotados em quadros. TORRES (2001)

O protocolo IP é não orientado à conexão, isto é, não verifica se o datagrama chegou ou não ao destino. Isso é feito pelo protocolo TCP, discutido a seguir.

De acordo com TORRES (2001), a principal função do IP é o roteamento, ou seja, adicionar mecanismos para que o datagrama chegue mais rapidamente possível ao seu

[3] *Media Access Control*. Endereçamento físico usado por cada interface de rede.

destino. Isso é feito com o auxílio dos roteadores da rede, que escolhem os caminhos mais rápidos entre a o destino.

DIÓGENES (2002) explica a comunicação na camada de rede da seguinte forma:

1- O protocolo IP recebe os segmentos vindos da camada de transporte e fragmenta-os em datagramas.
2- O protocolo IP do *host* de destino reagrupa estes datagramas de volta em segmentos e passa para a camada de transporte.

A Figura 2.11 mostra a estrutura do datagrama IP.

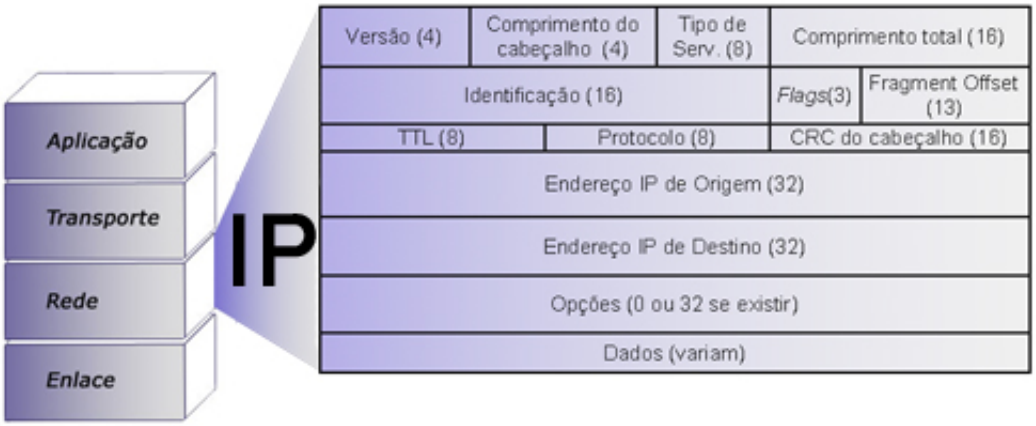


Figura 2.11: Estrutura do datagrama IP.

2.3.3 Camada de Transporte

Segundo TANENBAUM (1997), a finalidade da camada de Transporte é permitir que as entidades par (*peer entity*) dos *hosts* de origem e de destino mantenham uma conversação. Esta camada recebe os dados enviados pela camada de Aplicação e os transforma em pacotes, a serem repassados para a camada de Rede.

De acordo com TORRES (2001), a camada de Transporte utiliza um esquema de multiplexação, onde é possível transmitir “simultaneamente” dados das mais diferentes aplicações. O que ocorre, na verdade, é o conceito de intercalamento de pacotes: vários programas poderão estar se comunicando com a rede ao mesmo tempo, mas os pacotes gerados serão enviados à rede de forma intercalada, não sendo preciso terminar um tipo de aplicação de rede para então começar outra. Isso é possível graças ao uso do conceito de portas, já que dentro do pacote há informação da porta de origem e de destino do dado.

Nesta camada operam dois protocolos: o TCP (*Transmission Control Protocol*) e o UDP (*User Datagram Protocol*).

O UDP é um protocolo de transporte não orientado à conexão. Isso quer dizer que ele não verifica se o pacote de dados chegou ou não ao seu destino. Por essa razão, não é usado no transporte de dados importantes, como arquivos e *e-mails* (TORRES, 2001).

Assim, qualquer recurso necessário à verificação se os dados foram recebidos corretamente fica a cargo de uma aplicação externa, que passa a fazer o papel que normalmente é feito pelo protocolo TCP.

TORRES (2001) destaca como vantagem do UDP em relação ao TCP a velocidade, devido aos seguintes fatores:

- o tamanho do pacote de dados a ser transmitido fica menor, já que o cabeçalho UDP é bem menor que o cabeçalho TCP;
- no UDP não existe um mecanismo de verificação de chegada do pacote (*acknowledge*), que existe no TCP, acelerando o envio de pacotes, já que o transmissor não precisará esperar receber uma mensagem de *acknowledge* (reconhecimento) do receptor para enviar o próximo pacote.

O TCP é, segundo TORRES (2001), o mais complexo do sistema de protocolos TCP/IP. Ele recebe os datagramas IP e trata de colocá-los em ordem (já que em redes grandes os datagramas geralmente chegam fora de ordem) e verificar se todos chegaram corretamente.

Ao receber um quadro, a camada de Enlace da máquina receptora (o *driver* e a placa de rede) irá passar os dados para a sub-camada do protocolo IP, que por sua vez passará os dados para a sub-camada do protocolo TCP, que encaminhará para a aplicação correta. A camada TCP sabe para qual aplicação ela deve entregar os dados devido ao uso do conceito de portas.

A diferença ‘fundamental entre TCP e UDP é que o TCP é orientado à conexão. Portanto, para que o TCP possa operar, necessariamente deve-se ter uma conexão previamente estabelecida. Antes que um processo de aplicação possa começar a enviar dados a outros, os dois processos precisam enviar alguns segmentos preliminares um ao outro para estabelecer os parâmetros da transferência de dados em questão (KUROSE & ROSS, 2003).

Segundo COMER (1999), em TCP/IP: *Principles, Protocols and Architecture*, o protocolo TCP/IP cuida do fluxo de dados confiável entre dois *hosts*. Ele se preocupa com tarefas do tipo, repartir os dados que passam por ele vindos da camada de aplicação em pedaços de tamanho apropriado para a camada de rede, reconhecer pacotes recebidos ajustando *timeouts* para garantir o reconhecimento de pacotes que enviou, etc.

2.3.4 Camada de Aplicação

A camada de Aplicação é responsável pelo suporte das aplicações de rede. Existem vários protocolos que operam nesta camada. Os mais conhecidos são: o HTTP (*Hypertext Transfer Protocol*), o SMTP (*Simple Mail Transfer Protocol*), o FTP (*File Transfer Protocol*), o SNMP (*Simple Network Management Protocol*), o DNS (*Domain Name Service*) e o TELNET (Terminal Virtual) (TORRES, 2001).

Ainda de acordo com TORRES (2001), a camada de aplicação comunica-se com a camada de Transporte através de uma porta. As portas são numeradas e as aplicações padrão usam sempre uma mesma porta. Por exemplo: o protocolo SMTP utiliza sempre a porta 25, o HTTP utiliza sempre a porta 80 e o FTP usa as portas 20 (para transmissão de dados) e 21 (para transmissão de informações de controle).

O uso de um número de porta permite ao protocolo de transporte saber qual é o tipo de conteúdo do pacote de dados.

Geralmente a camada de Aplicação é um processo do usuário enquanto as outras três camadas são usualmente implementadas no sistema operacional. Outra diferença entre a camada de Aplicação e as outras três camadas é que esta se preocupa com os detalhes da aplicação e não com o movimento dos dados através da rede. As outras três camadas não sabem nada a respeito da aplicação, mas cuidam de todos os detalhes de comunicação.

2.4 Padrão de rede local - Ethernet

As redes locais são gerenciadas de acordo com as regras estabelecidas pelo **IEEE 802 *Standarts Committee* (***Institute of Electrical and Electronics Engineers*** –** organização fundada em 1963 que inclui engenheiros, estudantes e cientistas. Atua na criação e coordenação dos padrões e normas de comunicação e computação).

A Tabela 2.1 mostra algumas das normas do IEEE.

Tabela 2.1: Grupos de normas do IEEE.

Grupo	Função
802.2	Controle de Link Lógico (LLC)
802.3	Método de Acesso ao Meio CSMA/CD
802.3m	Fast Ethernet
802.3z	Gigabit Ethernet
802.5	Token Ring
802.1d	Protocolo Spanning Tree

O Ethernet é um padrão que define como os dados serão transmitidos fisicamente através dos cabos de rede. Dessa forma, essa arquitetura – assim como as arquiteturas Token Ring e FDDI – opera na camada de Enlace do modelo TCP/IP.

O padrão IEEE 802.3 é para uma LAN CSMA/CD1-persistente (*Carrier Sense Multiple Access with Collision Detection*). Para explicar: quando um *host* quer transmitir, ela escuta o cabo. Se o cabo estiver ocupado, o *host* aguarda até que ele fique livre; caso contrário, inicia imediatamente a transmissão. Se dois ou mais *hosts* começarem a transmitir simultaneamente em um cabo desocupado, haverá uma colisão. Todos os *hosts* que colidirem terminam sua transmissão, aguardam durante um tempo aleatório e repetem todo o processo novamente.

A Xerox foi quem iniciou o desenvolvimento da tecnologia Ethernet (o nome com referência ao éter luminífero através do qual se pensou que a radiação magnética se propagava). Junto com a DEC (*Digital Equipment Corporation*) e a Intel, criaram um padrão para um sistema Ethernet de 10 Mbps. Esse padrão formou a base do 802.3, que difere da especificação Ethernet por descrever uma família inteira de sistemas CSMA/CD (TANENBAUM, 1997).

Como diz TORRES (2001), o papel do Ethernet é, portanto, pegar os dados entregues pelos protocolos de alto nível TCP/IP, IPX/SPX, NetBEUI, etc. e inseri-los dentro dos quadros que serão enviados através da rede. O Ethernet também define como fisicamente esses dados serão transmitidos (o formato do sinal, por exemplo).

A transmissão usada no padrão Ethernet é banda-base, onde, de acordo com KUROSE & ROSS (2003), o adaptador envia um sinal digital diretamente ao canal

broadcast. A placa de rede não desloca o sinal para outra banda de freqüência, como é feito nos sistemas ADSL e *Cable Modem*. O padrão Ethernet usa a codificação Manchester, em que cada bit contém uma transição; um bit 1 tem uma transição de cima para baixo, ao passo que um 0 tem uma transição de baixo para cima. A operação de sincronização dos relógios dos adaptadores é possível através da codificação Manchester, que pode delinear cada bit e determinar se é 1 ou 0.

As redes Ethernet usam um método de acesso ao meio baseado em contenção e disputa do meio. Este método baseia-se no princípio de que apenas um dispositivo de rede pode usar o meio por vez; assim, os pontos de rede disputam o acesso ao meio. Um computador, ao tentar enviar sinal e notar que o mesmo esteja ocupado, se contém em enviar até que o outro computador termine a transmissão.

As três camadas do padrão Ethernet possuem as seguintes funções:

Camada Física: Transmite os quadros entregues pela camada de Controle de Acesso ao Meio usando o método CSMA/CD. Define como os dados são transmitidos através do cabeamento da rede e também o formato dos conectores usados na placa de rede.

Camada de Controle de Acesso ao Meio (MAC, IEEE 802.3): Monta o quadro de dados a ser transmitido pela camada física, incluindo cabeçalhos próprios dessa camada aos dados recebidos de Controle de Link Lógico.

Camada de Controle do *Link* Lógico (LLC, IEEE 802.2): Inclui informações do produto de alto nível que entregou o pacote de dados a ser transmitido. Com isso, a máquina receptora tem como saber para qual protocolo de alto nível ela deve entregar os dados de um quadro que ela acabou de receber.

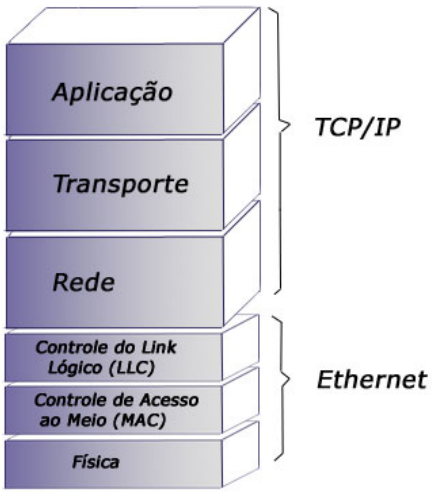


Figura 2.12: Sub-camadas da camada de Enlace no padrão Ethernet.

2.4.1 Camada Física

Em redes Ethernet, todos os micros compartilham o mesmo cabo, independentemente da topologia utilizada. Assim, o primeiro passo na transmissão de dados em uma rede Ethernet é verificar se o cabo está livre. Isso é feito pela placa de rede e daí o nome *Carrier Sense* (detecção de portadora) (TORRES, 2001).

Entretanto, o protocolo CSMA/CD não gera nenhum tipo de prioridade (daí o nome *Multiple Access*, acesso múltiplo). Com isso pode ocorrer de duas ou mais placas de rede perceberem que o cabo está livre e tentarem transmitir dados ao mesmo tempo. Quando isso ocorre, há uma colisão e nenhuma das placas consegue transmitir dados (TORRES, 2001).

Quando ocorre uma colisão, todas as placas de rede para de transmitir, esperam um período de tempo aleatório, e tentam a retransmissão. Como cada placa envolvida na colisão provavelmente gerará um valor aleatório diferente, possivelmente não ocorrerá outra colisão.

A existência de muitas máquinas em uma rede pode trazer uma perda de desempenho nessa rede. Mas não devemos pensar que há algo de errado com a rede devido às colisões. A colisão é um processo totalmente normal e desejável, já que faz parte do funcionamento do protocolo CSMA/CD. A queda de desempenho está mais ligada ao fato de somente uma máquina poder usar o cabo, impedindo as outras de transmitirem enquanto isso. Mas uma solução para isso, de acordo com TORRES (2001), é a divisão de uma rede Ethernet grande em diversas redes pequenas.

2.4.2 Camada de Controle de Acesso ao Meio

Endereçamento MAC

Em uma rede local, cada computador pode ser chamado de nó da rede. Entende-se, segundo KUROSE & ROSS (2003), um endereço de LAN como o endereço físico, endereço Ethernet ou endereço MAC (*Media Access Control* – Controle de acesso ao meio). E na verdade, não é o nó que tem um endereço de LAN e sim o adaptador de rede desse computador.

Esses endereços têm 6 bytes de comprimento, o que dá 2^{48} possíveis endereços de LAN. Cada fabricante recebe uma faixa de endereços de LAN e grava em seu adaptador o endereço de LAN garantindo assim que não haja dois adaptadores com o mesmo endereço.

O endereço de LAN de um adaptador tem uma estrutura linear (oposto da estrutura hierárquica) e jamais muda, não importando para onde vá o adaptador. Devemos lembrar que, contrastando com isso, um endereço IP tem uma estrutura hierárquica (isto é, uma parte que é da rede e outra do hospedeiro) (KUROSE & ROSS, 2003).

TORRES (2001), detalha que o endereço MAC é composto por 6 bytes, como mostra a Figura 2.13. Os três primeiros bytes são chamados OUI (*Organiationally Unique Identifier*) e identificam o fabricante da placa de rede e são padronizados pelo IEEE, isto é, para um fabricante deverá obrigatoriamente ser cadastrado no IEEE para conseguir número. Já os três últimos bytes são definidos e controlados pelo fabricante. Nada impede de um mesmo fabricante ter mais de um número OUI. Geralmente aqueles que produzem mais placas de rede têm um faixa de valores para serem usadas nessas placas.

Como os endereços MAC têm codificação hexadecimal, em que cada algarismo equivale a um número de quadro bits, um byte é representado por dois algarismos em hexadecimal e, com isso, o endereço MAC é sempre representado como um conjunto de 12 algarismos em hexadecimal. Exemplo: 11:22:33:44:55:66.

Figura 2.13: Estrutura do endereço MAC.

De acordo com TORRES (2001), quando um quadro é transmitido, todas as máquinas recebem este quadro ao mesmo tempo, já que todas estão conectadas a um mesmo cabo. Todas as placas de rede possuem o endereço MAC, que é gravado fisicamente dentro de uma memória ROM existente na placa. Com isso, mesmo todas as placas de rede recebendo o mesmo quadro ao mesmo tempo, somente a placa de rede cujo endereço MAC conste no campo Endereço MAC de Destino é que captura o quadro. A não ser quando se trata de um quadro de *broadcast*, que é recebido por todas as máquinas.

O papel primordial da camada de Controle de Acesso ao Meio (MAC) é gerar o quadro Ethernet, pegando os dados passados pela camada imediatamente superior a ela (Controle de Link Lógico, LLC) e acrescentando um cabeçalho a esses dados. Nesse quadro são inseridas as informações de qual placa de rede está enviando o quadro e para qual placa de rede o quadro está sendo enviado. Após gerar o quadro Ethernet, essa

camada envia o quadro para a camada Física, que é responsável pela transmissão desse quadro para o cabeamento da rede (TORRES, 2001).

Quando um adaptador quer enviar um quadro para algum adaptador de destino na mesma LAN, o adaptador remetente insere no quadro o endereço MAC do destino. Quando o adaptador de destino recebe o quadro, ele extrai o datagrama encerrado no quadro e o passa para cima da pilha de protocolos. Todos os outros adaptadores da LAN recebem também o quadro. Contudo, esses outros adaptadores descartam o quadro sem passar o datagrama de camada de rede para cima na pilha de protocolos. Assim, esses outros adaptadores não tem de interromper seus nós hospedeiros quando recebem datagramas destinados a outros nós. No entanto, às vezes um remetente quer que todos os outros adaptadores de LAN recebam e processem o quadro que ele está prestes a enviar. Nesse caso, o adaptador remetente insere um endereço broadcast especial no campo endereço do destinatário existente no quadro. Para LANs que usam endereços de 6 bytes (como a Ethernet), o endereço de broadcast é uma cadeia de 48 bits 1 consecutivos (FF-FF-FF-FF-FF-FF, em notação hexadecimal) (KUROSE & ROSS, 2003).

Quadro Ethernet (*Ethernet frame*)

A camada de Controle de Acesso ao Meio insere um cabeçalho aos dados recebidos da camada acima dela, formando o quadro Ethernet. A estrutura desse quadro pode ser vista na Figura 2.14. De forma resumida, segundo TORRES (2001), o quadro Ethernet possui um cabeçalho de 22 bytes, uma área de dados que varia entre 46 e 1500 bytes. Os campos do quadro Ethernet são:

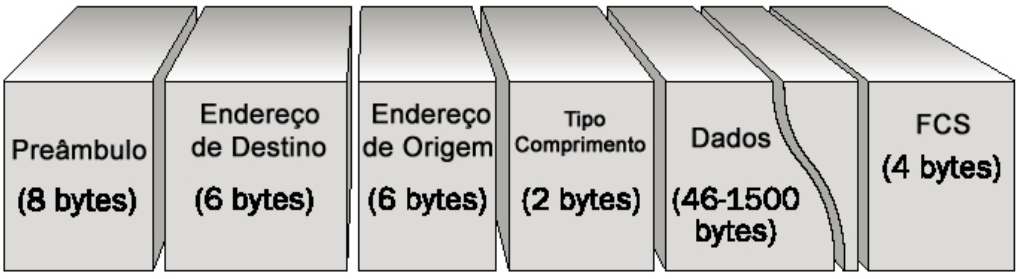


Figura 2.14: Estrutura do quadro Ethernet.

- **Preâmbulo:** Determina o inicio do quadro. São 7 *bytes* 10101010. Junto com o SFD

(*Start of Frame Delimiter*) forma um padrão de sincronismo, isto é, ao encontrar sete bytes 10101010 e um 10101011, o dispositivo receptor sabe estar diante do inicio de um quadro.

- SFD: É 1 *byte* 10101011.

- Endereço MAC de destino: o endereço físico ou MAC da placa de rede de destino. Um campo de 6 *bytes*. Se o endereço de destino for direcionado (*unicast*) neste campo haverá o endereço MAC do computador de destino. Caso seja um difusão (*broadcast*) o endereço físico que aparecerá será FFFFFFFFFFFF, pois a sinalização é em Hexadecimal.

- Endereço MAC de origem: o endereço físico ou MAC da placa de rede de origem. Também é um endereço MAC de 6 *bytes*. Neste campo não é possível haver endereços de *broadcast* ou *multicast* (difusão para um grupo de computadores).

- Comprimento ou Tipo: Indica quantos *bytes* estão sendo transferidos no campo de dados do quadro, já que o campo de dados de um quadro Ethernet tem tamanho variável e não fixo. Note que no frame 802.3 é usado o campo comprimento (*Lenght*). Já no frame Ethernet II usa a nomenclatura de tipo (*type*). Este campo tem como finalidade identificar o protocolo a ser usado na camada de rede.

- Dados: este campo contém as informações propriamente ditas a serem transmitidas. O campo pode variar entre 46 e 1500 *bytes*. São os dados enviados pela camada acima da camada de Controle de Acesso ao Meio.

***Pad*:** Se a camada de Controle do *Link* Lógico (LLC) enviar menos do que 46 *bytes* de dados para a camada de Contrele de Acesso ao Meio (MAC), então são inseridos dados chamados *pad* para completar os 46 *bytes*.

- Seqüência de checagem de frame (FCS): campo que tem como propósito armazenar o CRC, para o controle de correção de erros. Possui 4 *bytes*.

2.4.3 Controle do Link Lógico

O protocolo LLC pode ser usado sobre todos os protocolos IEEE do subnível MAC, como por exemplo, o IEEE 802.3 (Ethernet), IEEE 802.4 (*Token Bus*) e IEEE 802.5 (*Token Ring*).

Uma funcionalidade da camada de Controle do Link Lógico, de acordo com TORRES (2001), é receber os dados repassados pelo protocolo de alto nível instalado na máquina (TCP/IP, IPX/SPX, NetBEUI, etc.) e acrescentar justamente a informação de qual

protocolo foi responsável por gerar os dados. Com isso, torna-se possível o uso simultâneo de vários protocolos de alto nível em uma mesma máquina.

Usa-se o LLC quando é necessário controle de fluxo ou comunicação confiável. Ele oferece três opções de transmissão: serviço de datagrama não-confiável, serviço de datagrama com confirmação e serviço orientado à conexão confiável. O LLC consegue isso dividindo a mensagem a transmitir em quadros com algumas centenas de *bytes* de dados e alguns *bytes* de controle (como CRC, por exemplo). Enquanto transmite seqüencialmente os quadros de dados, o transmissor deve tratar os quadros de reconhecimento (ACK), que são enviados pelo receptor a fim de indicar se a transmissão ocorreu com ou sem erros. Caso algum quadro não tenha chegado corretamente, o transmissor deve retransmiti-lo, e o receptor deve descartar o quadro errado.

Um ruído mais forte no cabeamento pode destruir completamente um quadro. Nesse caso, os protocolos da camada de enlace devem retransmitir essa informação. Entretanto, múltiplas retransmissões do mesmo quadro podem fazer com que existam quadros duplicados. Um quadro duplicado pode acontecer se, por exemplo, o ACK do receptor foi destruído. É tarefa da camada de Controle de Link Lógico (LLC) tratar e resolver problemas causados por quadros danificados, perdidos e duplicados.

Outra função do sub-nível de enlace LLC é controle de fluxo, ou seja, o controle de um transmissor rápido para que não inunde de dados um receptor mais lento. Algum mecanismo regulador de tráfego deve ser empregado para deixar o transmissor saber quanto espaço em *buffer* tem no receptor naquele momento. Freqüentemente, o controle de fluxo e de erro é integrado, simplificando o protocolo.

Para entender quando é necessário controle de fluxo, suponha um transmissor que pode enviar dados a 1Mbps, e um receptor que pode receber dados somente a 100Kbps, como mostra a Figura 2.15. Evidentemente, algum controle deve haver para que o receptor não seja obrigado a descartar dados.

Figura 2.15: Fluxo entre transmissor e recptor.

Outra complicação que deve ser tratada em nível de enlace é quando a linha for utilizada para transmitir tráfego em ambas as direções (de A para B e de B para A). Normalmente, uma comunicação envolve a transmissão do pacote de dados e o ACK (*acknowledge*) enviado de volta pela estação receptora, indicando que os dados chegaram sem erros. Entretanto, o problema é que os quadros de ACK competem pelo meio físico da mesma forma que os quadros de dados, prejudicando o desempenho do sistema. Para eliminar esse problema, em alguns protocolos utiliza-se o conceito de *piggybacking*, onde os *bits* de ACK que devem ser enviados em resposta ao quadro de dados transmitidos pela estação **A** vem junto com o quadro de dados que a estação **B** quer transmitir para a estação **A**.

Resumindo, as principais funções do nível de enlace são as seguintes:

- Entregar ao nível de rede os dados livres de erros de transmissão;
- Retransmissão de quadros errados;
- Controle de fluxo;
- Tratamento de quadros duplicados, perdidos e danificados.

Quanto à função de receber os dados repassados pelo protocolo de alto nível e acrescentar a informação de qual protocolo foi responsável por gerar os dados, citada por TORRES (2001), a implementação padrão da camada de Controle do Link Lógico (LLC) adicionava um cabeçalho de apenas 3 bytes aos dados recebidos do protocolo de alto nível: DSAP (*Destination Service Access Point*, que indica o protocolo de destino), SSAP (*Source Service Access Point*, que indica o protocolo de origem) e Controle.

Essa implementação mostrou-se ineficaz para identificar corretamente o protocolo de origem e o de destino, isto é, havia poucos bytes para ser feita corretamente essa identificação – ainda mais porque o número que cada protocolo usaria não era muito bem padronizado.

A solução para essa deficiência foi a criação de um campo SNAP (Sub Network Access Protocol), de cinco bytes, usando um padrão IEEE para protocolos. Nesse padrão, três bytes são usados para identificar o fabricante/desenvolvedor do protocolo e os outros dois bytes são definidos pelo fabricante/desenvolvedor internamente, assim como ocorre no endereço MAC.

2.5 Domínios de Colisão e Difusão

Segundo DIÓGENES (2002), o método de acesso ao meio e as diferentes formas que os equipamentos de conectividade têm para filtrar o tráfego na rede e as formas de transmissão de sinais no meio são divididas em:

Domínio de colisão: quando temos uma rede cuja segmentação é feita no nível de enlace. Por exemplo, com um *switch* que liga dois segmentos, temos dois domínios de colisão. Com o uso do *hub* (que não filtra endereço MAC), temos apenas um domínio de colisão.

Domínio de Broadcast: método baseado na camada de rede, onde acontecem os roteamentos. Com isso só haverá mais de um domínio de *broadcast* se houver um roteador na rede. Por exemplo, um roteador de três portas são três domínios de *broadcast*. Já um switch de quatro segmentos é apenas um domínio de *broadcast*. O domínio de *broadcast* também acumula o domínio de colisão, ou seja, em um roteador de três portas temos também três domínios de colisão.

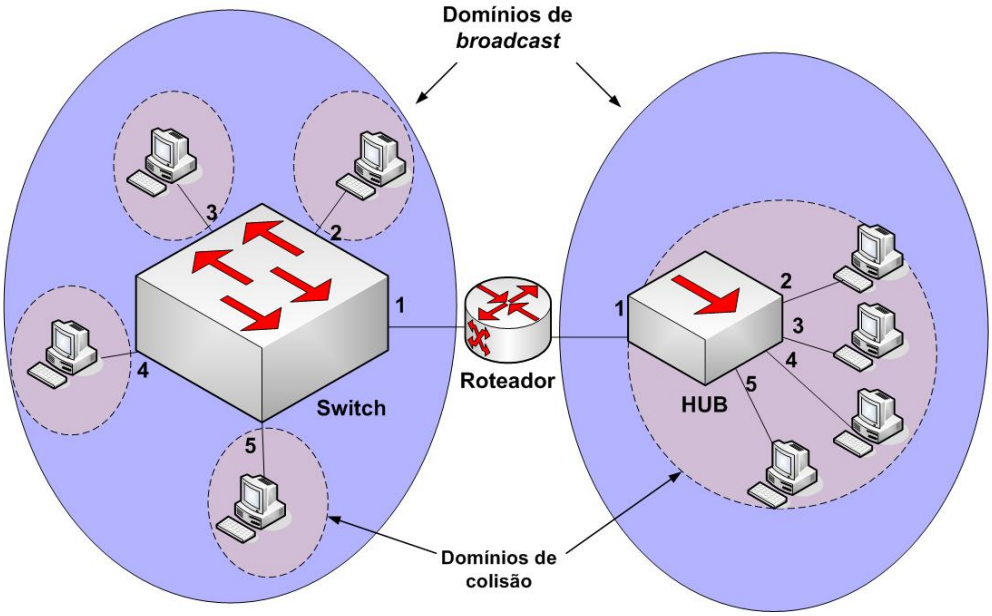


Figura 2.16: Domínios de difusão e colisão.

2.6 Equipamentos de Interconexão

As instituições – incluindo empresas, universidades e escolas – são, em geral, constituídas de muitos departamentos; cada departamento tem e administra sua própria LAN Ethernet. Por isso, é de grande importância que seus departamentos, salas e setores interconectem suas redes locais.

Nesse contexto, existem três equipamentos para executar a função de interconectar os diferentes departamentos de uma instituição: os H*ubs*, *Switches*/*Bridges* e os roteadores.

2.6.1 Hub (Repetidor)

O *hub* é um dispositivo simples que toma uma entrada (os bits de um quadro) e a retransmite para suas portas de saída. Os *hubs* são, essencialmente, repetidores que operam com *bits*, que trabalham na camada mais baixa (KUROSE & ROSS, 2003).

Quando um *bit* chega a uma interface no *hub*, este simplesmente transmite o *bit* de maneira simultânea para todas as outras interfaces (portas). DIÓGENES (2001) ressalta que um *hub* não tem inteligência suficiente para diferenciar um endereço de origem e destino dentro de um quadro (*frame).*

Na Figura 2.17 pode ser observado um projeto de *hub* multinível, usado em redes departamentais, em que o arranjo é feito de forma hierárquica. Em um projeto multinível, há uma rede interconectada como uma LAN e parcelas departamentais da LAN (cada *hub* departamental e as estações conectadas a ele) – chamadas de segmentos de LAN.

De acordo com KUROSE & ROSS (2003), nesse exemplo de rede (departamental), todos os segmentos pertencem ao mesmo domínio de colisão, isto é, sempre que uma ou mais estações dos segmentos de LAN transmitem ao mesmo tempo, há colisão.

Embora haja muitos benefícios em se conectar estações através de um *hub*, podemos destacar algumas limitações, que dificultam sua disseminação:

- o fato de quando as LANs interconectadas por um *hub* fazer que com que os domínios de colisão dos departamentos, por exemplo, se transformem em um grande domínio de colisão comum;
- não podem conectar diferentes tecnologias Ethernet. Isto torna impossível conectar estações que trabalham em 10BaseT com outras que usam 100BaseT;
- divide a largura de banda proporcionalmente para cada porta do *hub*, isto é, cada uma das 24 portas de um *hub* de 100 Mbps estará trabalhando a 4.5 Mbps, caso todas estejam sendo usadas ao mesmo tempo.

Segundo DIÓGENES (2001), pelo fato do hub compartilhar o meio e funcionar apenas como um repetidor de sinais, torna-se notável que a adição de mais computadores ao hub irá decrementar a performance da rede como um todo.

Hoje em dia, os *switches* já estão substituindo de forma gradativa os *hubs*, otimizando ainda mais a comunicação na rede local. *Switches* serão detalhados nos capítulos seguintes.

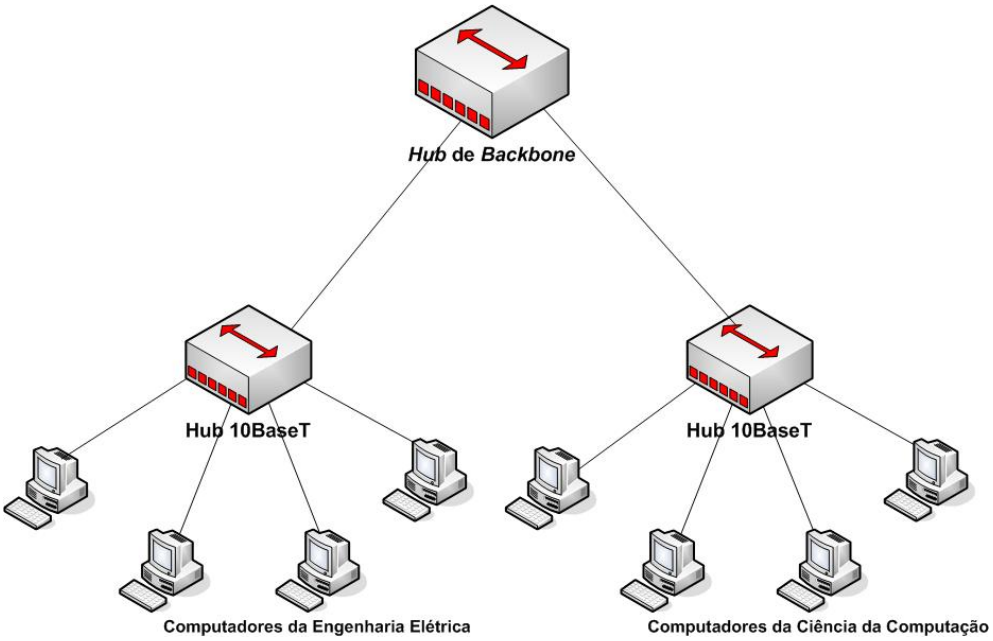


Figura 2.17: LANs interconectadas por *Hubs.*

2.6.2 Ponte (*Bridge*)

Diferentemente dos *hubs*, que não são capazes de analisar quadros, as *bridges* ou pontes operam sobre os quadros Ethernet (camada de Enlace), repassando e filtrando estes quadros usando os endereços de LAN de destino. KUROSE & ROSS (2003) explicam seu funcionamento:

"Quando um quadro chega a uma interface de bridge, esta não copia o quadro para todas as outras interfaces. Em vez disso, a bridge examina o endereço de destino do quadro (endereço MAC) e tenta repassá-lo para a interface que leva a esse destino."

No contexto da Figura 2.18, podemos observar duas redes interconectadas por meio de *bridges*. Os números dos segmentos são as portas da bridge usadas pelos *hubs*. Assim, em contraste com o projeto de hub multinível da Figura 2.17, cada segmento da Figura 2.18 fica isolado em um domínio de colisão.

KUROSE & ROSS (2003) enfatizam que as *bridges* vieram para suplantar muitos problemas que atormentam os *hubs*. Elas permitem comunicação entre redes diferentes, preservando, ao mesmo tempo, domínios de colisão isolados para cada uma. Também

podem interconectar diferentes tecnologias LAN, como 10Mbps e 100Mbps, além de não ter a limitação ao tamanho possível de uma LAN, podendo ligar quantas redes forem necessárias.

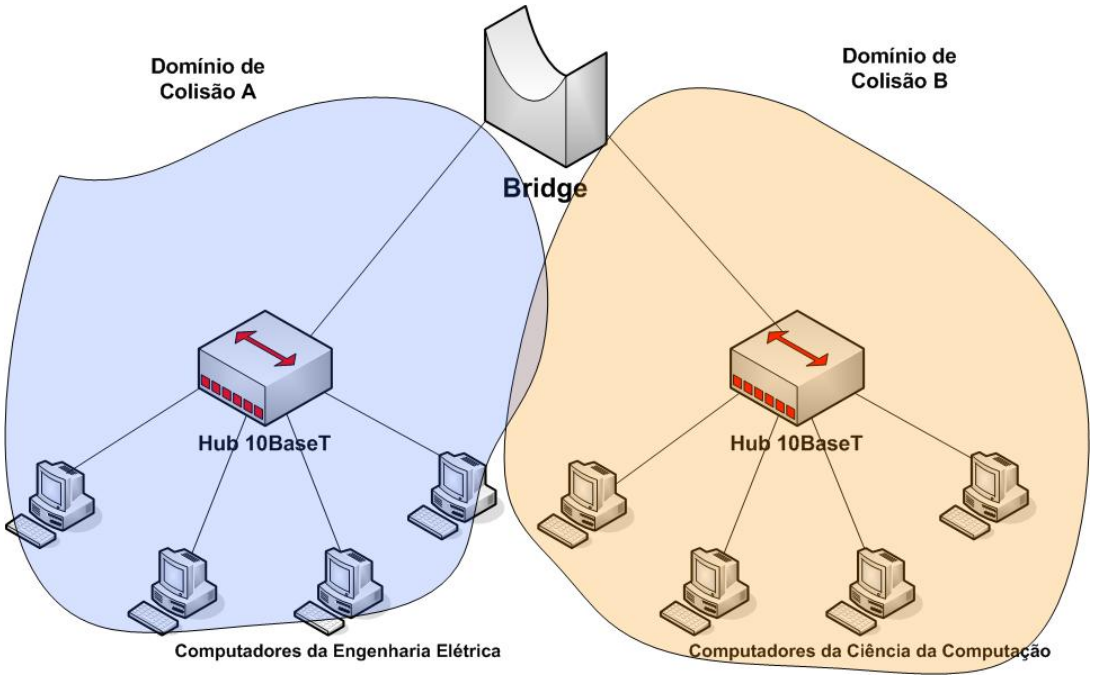


Figura 2.18: Segmentação da redes pela *Bridge*.

Repasse e filtragem pelas *bridges*

Filtragem é a capacidade de uma *bridge* determinar se um quadro deve ser repassado para alguma interface ou se deve apenas ser descartado. Repasse é sua capacidade de determinar se as interfaces para as quais um quadro deve ser dirigido. A filtragem e o repasse são feitos com uma **tabela de *bridge***.

Fazendo-se das palavras de KUROSE & ROSS (2003), podemos dizer que a tabela de *bridge* contém registros para alguns nós da LAN, mas não necessariamente todos. O registro de um nó na tabela de *bridge* contém:

1 – o endereço de LAN do nó (MAC);

2 – a interface (porta) da bridge que leva em direção ao nó;

3 – o tempo em que o registro para o nó foi colocado na tabela.

É importante lembrar que a montagem da tabela de *bridge* é muito diferente da montagem da Tabela de roteamento.[4]

[4] Tabela usada pelo roteador, ao receber um pacote, para verificar se contém uma rota para a rede de destino. Pode ser uma rota direta ou então para qual roteador o pacote deve ser enviado.

2.6.3 Comutadores (*Switches*)

Comutadores ou *Switches* são, em essência, *bridge* multi-interface (multiportas) de alto desempenho. Tal como as *bridges*, eles repassam e filtram quadros usando endereços MAC e montam as tabelas de repasse automaticamente (auto-aprendizagem). A diferença mais importante entre *bridge* e *switch*, segundo KUROSE & ROSS (2003), é que *bridges* comumente têm um número menor de interfaces (geralmente duas ou quatro), enquanto os *switches* podem ter dezenas delas. Um grande número de interfaces gera uma alta velocidade de transmissão agregada por meio do elemento comutador, necessitando, consequentemente, de uma arquitetura de alto desempenho.

Um switch faz a comunicação múltipla entre os computadores conectados a ela, ao contrário do hub em que apenas um por vez pode transmitir.

DIÓGENES destaca que segurança é um ponto importante que se ganha com o uso de switches. Em uma rede baseada em hub todos os dados estão disponíveis para todas as estações conectadas ao segmento. Qualquer pessoa com acesso ao hub e com um software específico poderia monitorar e capturar dados que estivessem trafegando no meio. Isto não é possível com uma LAN baseada em *switch* porque a origem envia dados para o destino de forma direcionada (*unicast*) e não se torna disponível para os outros computadores conectados ao mesmo *switch*.

Através de switches, assim como *bridges*, podemos segmentar a rede no nível de enlace do modelo TCP/IP. A segmentação acontece através da detecção do endereço MAC. *Switches* aprendem os endereços à medida que são trocadas informações entre os computadores da rede.

A divisão da rede em segmentos de colisão ou domínios de colisão é uma tarefa também realizada pelos *switches*, porém a rede como um todo continuava sendo um único e grande domínio de *broacast*. *Switches* não tem a capacidade de dividir a rede em diferentes domínios de *broadcast*. Isso só pode ser realizado na camada de Rede do modelo TCP/IP, através de roteadores.

De acordo com DIÓGENES, são aspectos que devemos considerar, quanto à funcionalidade de *switches* e *bridges*:

- as *bridges* são implementadas via *software* enquanto *switches* são baseados em *hardware*, pois usam um *chip* baseado em ASICs (*Application-Specific Integrated Circuits*) para auxiliar nas decisões de filtragem;

- uma *bridge* só pode ter uma instância *Spanning Tree*, que é tema deste trabalho, enquanto *switches* podem ter várias.

KUROSE & ROSS (2003) ainda destacam duas características importantes dos switches, a saber:

No modo *full-duplex* os *switches* podem enviar e receber quadros ao mesmo tempo sobre a mesma interface. Com um *switch full-duplex* (e correspondentes adaptadores Ethernet *full-duplex* nos *hosts*), o *host* A pode enviar um arquivo ao *host* B, enquanto o *host* B envia simultaneamente ao *host* A. Isso proporciona uma interessante propriedade aos *switches*: o adaptador do *host* pode transmitir (e receber) quadros à velocidade de transmissão total de seu adaptador. Em particular, o adaptador do hospedeiro sempre percebe um canal ocioso e nunca sofre uma colisão. Quando um computador tem uma ligação direta com um *switch*, dizemos que ele tem acesso dedicado. Ou seja, permite que simultâneas transações de pacotes entre os *hosts* que estão ligados ao *switch*. (Figura 2.19)

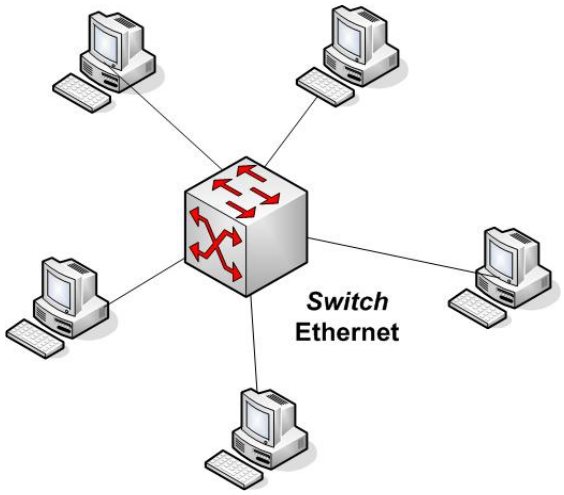


Figura 2.19: *Switch* Ethernet fornecendo acesso dedicado a cinco *hosts.*

De forma geral, DIÓGENES caracteriza *switches* em três aspectos:

- Leitura de Endereços: os *switches* aprendem os endereços MAC de acordo com a comunicação existente na rede e com isso monta uma tabela a qual vai crescendo à medida que os computadores trocam informações entre si. Quando um *switch* é iniciado, sua tabela de endereçamentos MAC está vazia. Quando um nó de rede transmite um sinal este é recebido pelo *switch*, através da porta à qual está conectado o dispositivo, e então captura o endereço de origem e adiciona à tabela, vinculando este MAC à porta (*backward learning*).

À medida que outros computadores originam transmissões de dados esta tabela vai sendo preenchida, e no caso de haver dois endereços MAC na tabela, já é possível realizar uma transmissão ponto a ponto entre os dois nós da rede.

- Decisões de Encaminhamento e Filtragem: *switches* filtram o quadro recebido pelo endereço MAC, não permitindo que o quadro seja enviado para todos os computadores, através da tabela de endereçamentos MAC. Esta tabela que é montada sob demanda também é utilizada para que o *switch* possa tomar decisões de encaminhamento e filtragem dos quadros. Quando o quadro passa por uma porta do *switch* o endereço MAC de destino é comparado com a base de dados de Encaminhamento e Filtragem de MAC. Desta forma se o endereço de destino está nesta base de dados o quadro é enviado apenas para a devida porta listada. Isto, além de prevenir o congestionamento da rede, também traz uma melhoria na largura de banda disponível. Caso o endereço de destino contido no quadro Ethernet não esteja nesta tabela, o quadro será transmitido para todas as outras portas exceto a porta que o originou.

- Prevenção de *loops*: se na rede existir uma ligação entre múltiplos *switches*, seja para oferecer redundância, seja para expandir a rede, é possível que venham a existir *loops*, que são indesejáveis em uma rede. Para prevenir isso, *switches* utilizam o protocolo STP (*Spanning Tree Protocol*). Cenários de implementação de *switches* com links redundantes entre esses *switches* são uma ótima prática, já que garante uma maior disponibilidade de acesso.

A Figura 2.20 esquematiza a implementação de enlaces redundantes através de *switches*.

DIÓGENES nos leva a compreender que, em contrapartida ao uso de múltiplos *links*, que em alguns casos são extremamente usados, eles também podem causar alguns erros que podem ser muito prejudiciais à rede, são alguns deles:

- Difusão de *broadcasting*: se não houver nenhum tipo de prevenção de *loop* na rede, um *broadcast* pode tomar uma dimensão de tráfego que vai decrementar sensivelmente a performance de toda a rede. Isso acontece porque o uso de *switches* para interligar redes cria um único domínio de *broadcast* ou domínio de difusão.

- Múltiplas cópias de quadros: como o quadro pode ir por vários caminhos para chegar ao destino, uma cópia de múltiplos quadros, ao chegar no *switch*, vai causar um resultado

inesperado, pois a consulta na tabela de endereços MAC ficará confusa na hora de escolher para onde encaminhar o quadro.

- Múltiplos *loops*: um *loop* de quadro pode acontecer dentro de um outro *loop*, pois não haverá como a *switch* fazer suas tarefas de comutação de quadros.

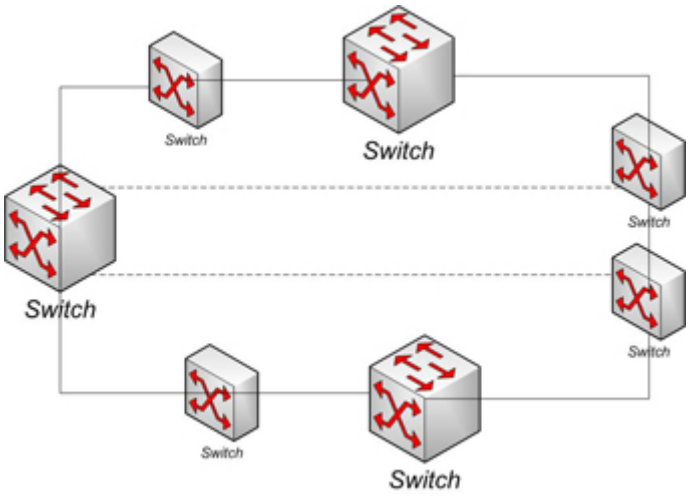


Figura 2.20: Cenário de múltiplos caminhos para garantir alta disponibilidade.

O objetivo de um *switch* Ethernet é, de acordo com FURTADO (2003), oferecer a micro-segmentação, onde o tráfego entre dois *hosts* deverá ocorrer somente entre as duas portas que conectam ambos, devendo as demais portas ficarem isentas do recebimento de pacotes pertinentes à esta conversação. Um *switch* Ethernet utiliza a porção MAC de origem do *frame* Ethernet para aprender o endereço MAC de um computador, mapeando-o para a respectiva porta do *switch*, e a porção MAC de destino deste frame para consultar a sua tabela de endereçamento MAC e permitir o encaminhamento do tráfego somente para a porta necessária para o recebimento destes pacotes. Entretanto, há ocasiões onde um *switch* Ethernet efetuará o encaminhamento do tráfego para todas as portas, exceto à porta de origem, e isto corre quando há:

- ***Unknown Unicast*:** Ao receber um frame, o switch faz uma pesquisa em sua tabela, tendo como base o endereço MAC de destino informado pelo *frame*. Caso não haja nenhuma entrada correspondente na tabela, o *frame* é encaminhado para todas as portas do *switch*, exceto aquela que originou o pacote. Este tipo de situação é comum quando conectamos um *host* pela primeira vez no *switch*, e o *switch* ainda não possui a informação sobre que porta este novo computador está conectado. Quando este computador enviar o seu primeiro pacote pela rede, o *switch* aprenderá o seu endereço MAC e fará o devido mapeamento e inserção

desta informação em sua tabela. *Unknown Unicasts* costumam ocorrer com freqüência, também, quando temos um sistema que fica tentando a comunicação com um *host* inexistente para o *switch*;

- ***Broadcast*:** Devido à natureza de um endereço *broadcast* na camada de enlace (0xFFFFFFFF), fica impossível mapearmos uma porta para este endereço. B*roadcast* significa "todos" e, neste caso, o *switch* deverá encaminhar pacotes *broadcast* para todas as portas, exceto à porta de origem;
- ***Multicast***: Uma vez que somente o campo MAC de origem de um quadro é utilizado para aprender a localização física (porta do *switch*) de um determinado computador, e que o endereço *multicast* na camada de enlace é sempre utilizado como endereço MAC de destino, o *switch* jamais terá condições de inserir um endereço *multicast* em sua tabela. Isso torna impossível, portanto, efetuar o mapeamento "MAC x porta". Sendo assim, pacotes para endereços *multicast* serão encaminhados para todas as portas, exceto à porta de origem (FURTADO, 2003).

2.6.4 Roteadores

Um roteador é um equipamento de interconexão entre redes responsável por tomar a decisão de qual caminho os dados deverão seguir. Roteadores não analisam os quadros físicos que estão sendo transmitidos, mas sim os datagramas produzidos pelo protocolo IP de alto nível. No caso do TCP/IP, os roteadores são capazes de ler e analisar os datagramas IP contidos nos quadros transmitidos pela rede (TORRES, 2001).

Ao mesmo tempo em que os roteadores ajudam na implementação de redes fisicamente dispersas entre eles, também influenciam em outros pontos que podem ser considerados como um gargalo na rede. São eles, de acordo com DIÓGENES (2002):

- Redundância;
- Número de hosts por segmento;
- Topologias dissimilares de rede.

A grande diferença entre *switches* e roteadores é que o tipo de endereçamento. O utilizado pelo *switch* ou *bridge* é o endereçamento usado na camada de Enlace, ou seja, o endereço MAC das placas de rede, que é um endereço físico. O roteador, por operar na

camada de Rede, usa o sistema de endereçamento dessa camada, que é um endereçamento lógico. No caso do TCP/IP, esse endereçamento é o endereço IP (TORRES, 2001).

Em redes grandes – a Internet é o melhor exemplo – é praticamente impossível para um *switch* saber os endereços MAC de todas as placas de rede existentes na rede. Quando um *switch* não sabe o endereço MAC de alguma máquina, ela usa a técnica de inundação (*flooding*), isto é, envia o pacote de dados para todas as suas portas. Isso em redes de maior porte é impraticável, visto que a inundação gerada por uma rede trabalhando unicamente com *switches* seria excessiva. TORRES (2001) afirma que é por esse fato que os roteadores operam com os endereços lógicos – que trabalham em uma estrutura onde o endereço físico não é importante – e a conversão desse endereço (endereço IP) para o endereço físico (endereço MAC) é feita somente quando o datagrama chega à rede de destino.

Ainda segundo TORRES (2001), a vantagem do uso de endereços lógicos em redes grandes é que eles são mais fáceis de serem organizados hierarquicamente, de uma forma padronizada. Mesmo que um roteador não saiba onde está fisicamente localizada uma máquina que possua um determinado endereço, ele envia o pacote de dados para um outro roteador que tenha probabilidade de saber onde esse pacote deve ser entregue (roteador hierarquicamente superior). Esse processo continua até o pacote atingir a máquina de destino.

Entre todas as características, a mais importante é a proteção contra as tempestades de *broadcast* na camada de Enlace. Este é um problema traumático que ocorre constantemente em redes grandes, como por exemplo as redes Campus, em que quadros de difusão (*broadcast*) chegam a *hosts* que não estão participando da comunicação, prejudicando assim o tráfego daquela rede. Os roteadores fornecem isolamento de tráfego mais robusto, controlam essas tempestades de *broadcast* e usam rotas ‘mais inteligentes’ entre os *hosts* da rede (KUROSE & ROSS, 2003).

2.7 Redundância

KUROSE & ROSS (2003) nos explicam que um dos problemas de um projeto hierárquico puro para segmentos de LAN interconectados é que, se um *switch* próximo ao topo da hierarquia cair, os enlaces da LAN ficarão desconectados. Por essa razão, é

recomendável montar redes com múltiplos trajetos entre os enlaces dessa rede. Um exemplo disso é mostrado na Figura 2.21.

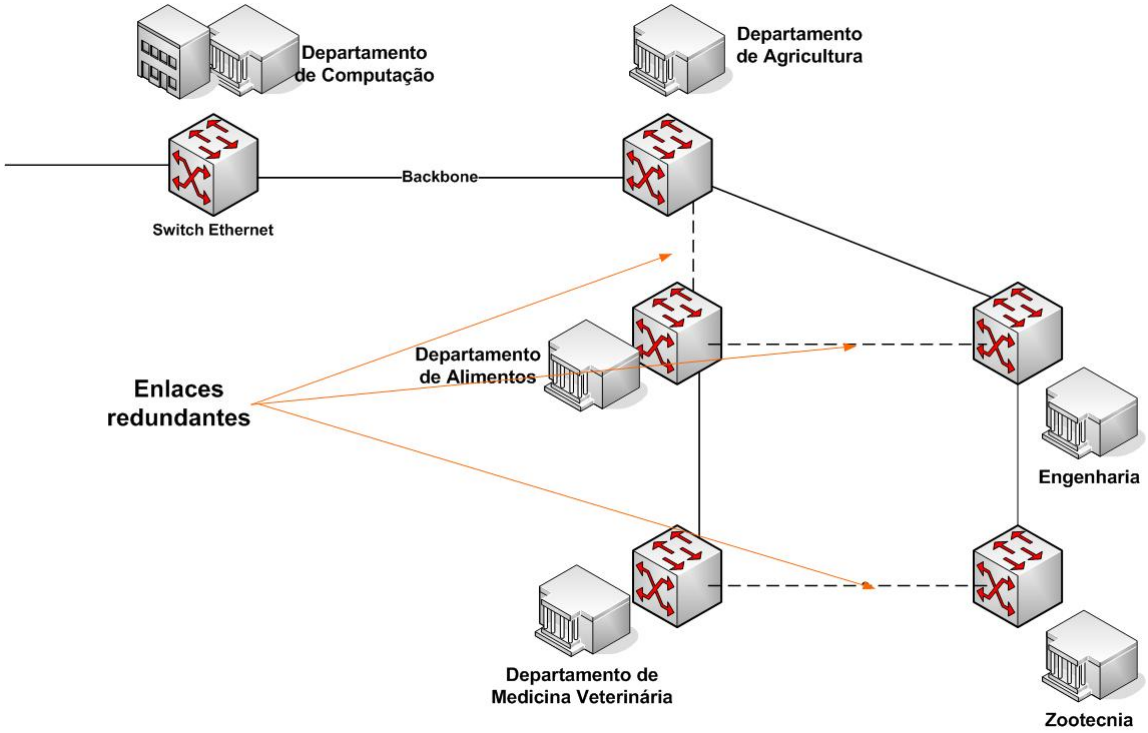


Figura 2.21: Enlaces redundantes em uma rede.

Segundo PINHEIRO (2004), o projeto bem sucedido de uma rede de computadores pode ser representado pela capacidade desta em oferecer os serviços essenciais requeridos por seus usuários e por preservar os seus principais componentes na eventual ocorrência de falhas.

A fim de prevenir eventuais falhas e oferecer alternativas que evitem que estas acarretem maiores prejuízos, se faz necessário que os projetos contemplem planos de redundância e contingência constituídos por uma série de ações e procedimentos que visam soluções e dispositivos de recuperação relacionados com essas falhas.

2.7.1 Falhas

No ambiente das redes de computadores podemos destacar vários aspectos críticos que podem ser considerados pontos de falhas potenciais para o sistema: cabeamento, servidores, subsistemas de disco, entre outros. Nesse contexto, as falhas são consideradas como eventos danosos, provocados por deficiências no sistema ou em um dos elementos internos dos quais o sistema dependa.

As falhas podem ser derivadas de erros no projeto do *software*, degradação do hardware, erros humanos ou dados corrompidos. Entretanto, só existem duas variáveis para a paralisação temporária de uma rede em função de condições de falha que não se podem definir ou prever:

- **Indisponibilidade –** Corresponde ao período de inatividade ou "*downtime*" da rede (programado ou não). As características do projeto devem ser suficientes para garantir que a informação seja replicada automaticamente do ambiente de produção para o ambiente de contingência, de forma que o tempo de indisponibilidade do sistema seja reduzido, melhorando o nível de serviço e atendendo às exigências dos usuários;
- **Instabilidade –** é imprescindível conhecer quais são os parâmetros considerados como normais dentro do ambiente. A correta definição de métricas de qualidade, bem como a implantação de mecanismos de coleta e controle de variáveis do sistema são imprescindíveis para a configuração de ações de correção imediatas e de análises de tendências.

2.7.2 Provendo Redundância na Topologia de Rede

Uma forma de construir redes altamente disponíveis é provendo segurança e confiabilidade através da redundância na topologia da rede preferencialmente aos equipamentos de rede. Na rede Campus da Figura 2.22, um caminho de *backup* existe entra cada *switch*.

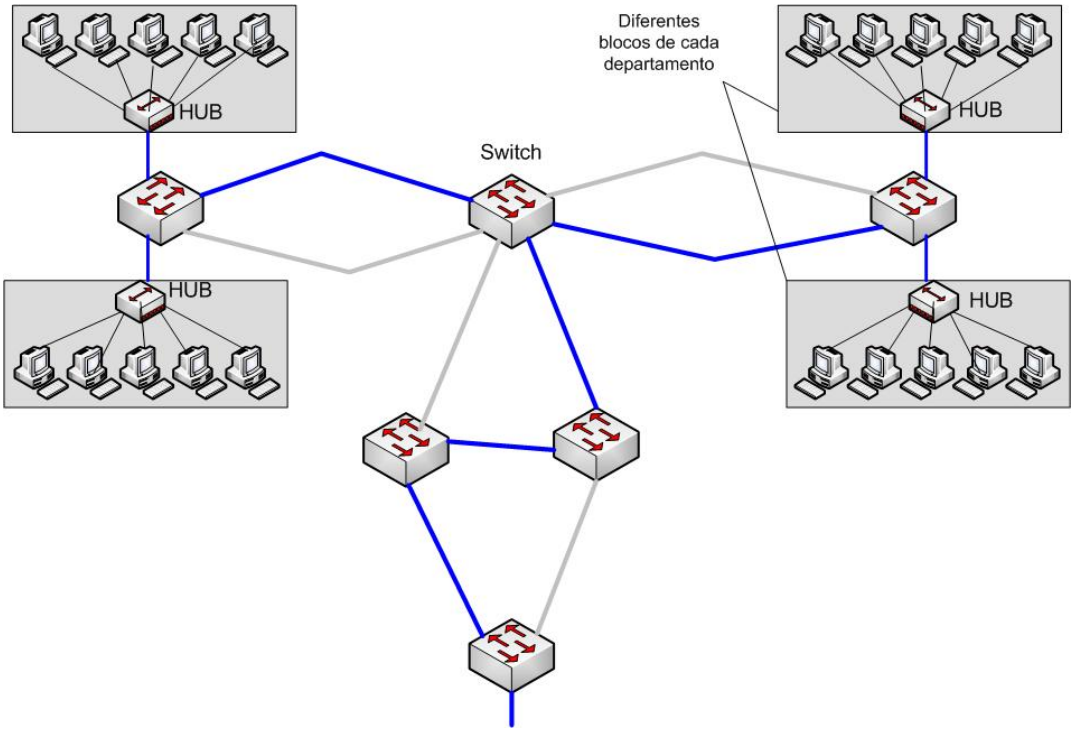


Figura 2.22: Rede com caminhos redundantes.

Apesar do uso de caminhos redundantes e dispositivos de segurança aumentarem os custos e serem mais difíceis de gerenciar, oferecem uma série de vantagens:

- Os equipamentos que oferecem redundância precisam ser configurados paralelamente aos elementos principais da rede. Esta prática reduz a probabilidade de que os problemas com o ambiente físico interromperão o serviço.
- A indisponibilidade temporária de um equipamento, em conseqüência de alguma falha, não interrompe os serviços e mantém a rede operante por um tempo muito maior.

Trajetos múltiplos redundantes entre os segmentos de LAN (como as LANs departamentais de uma universidade) podem melhorar muito a tolerância à falha. Mas, infelizmente, trazem sérios efeitos colaterais – os quadros podem circular e se multiplicar dentro da LAN interconectada (KUROSE & ROSS, 2003).

DIÓGENES (2001) nos dá mais detalhes sobre esses problemas, que prejudicam consideravelmente o desempenho de uma rede:

- Difusão de *broadcasting*: se não houver nenhum tipo de prevenção de *loop* na rede, um *broadcast* pode tomar uma dimensão de tráfego que vai decrementar sensivelmente a

performance de toda a rede. Isso acontece porque o uso de *switches* para interligar redes cria um único domínio de *broadcast* ou domínio de difusão. Este problema também é conhecido como Tempestade de *Broadcast*.

- Múltiplas cópias de quadros: como o quadro pode ir por vários caminhos para chegar ao destino, uma cópia de múltiplos quadros, ao chegar no *switch*, vai causar um resultado inesperado, pois a consulta na tabela de endereços MAC ficará confusa na hora de escolher para onde encaminhar o quadro.

- Múltiplos *loops*: um *loop* de quadro pode acontecer dentro de um outro *loop*, Assim não haverá como o *switch* realizar suas tarefas de comutação de quadros.

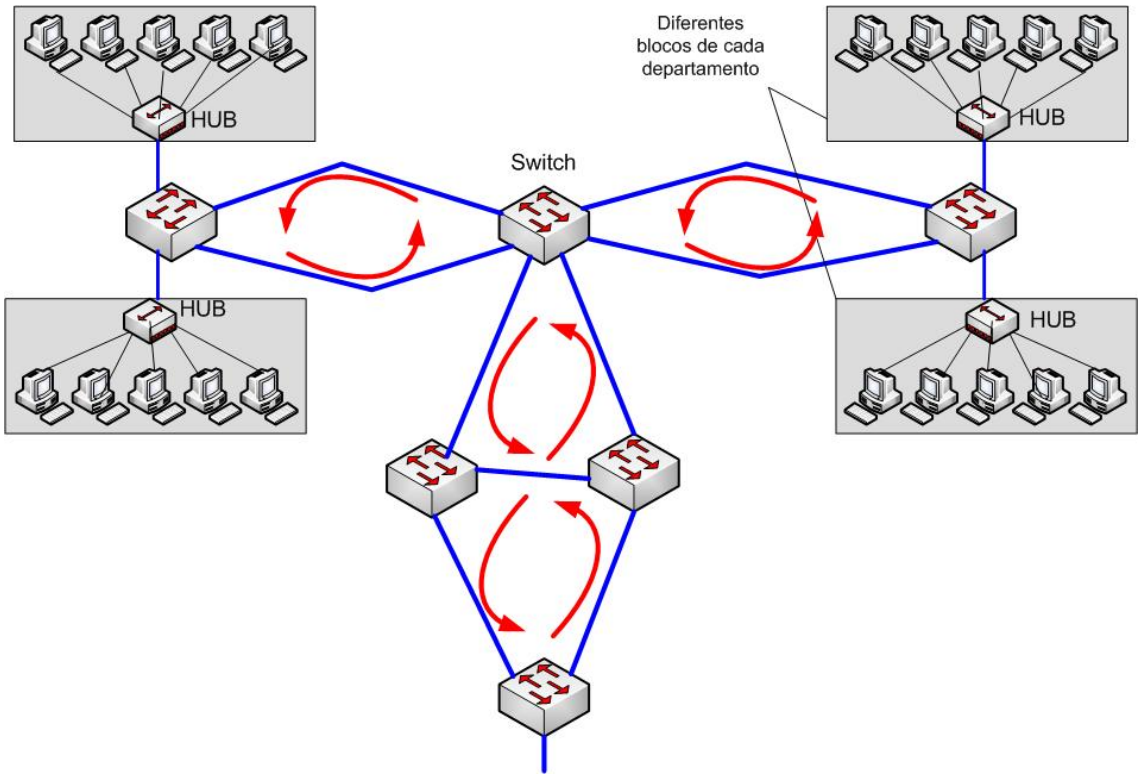


Figura 2.23: *Loops* em uma rede com enlaces redundantes.

Para solucionar o problema dos loops e das tempestades de *broadcast*, *switches* podem fazer uso da tecnologia *Spanning Tree Protocol*, detalhado na próxima seção.

2.8 *Spanning Tree Protocol* (IEEE 802.1d)

Quando projetos de redes de computadores utilizam múltiplos *switches*, segmentos Ethernet redundantes, ou seja, *links* alternativos são frequentemente usados entre estes *switches*. O objetivo é simples. Os *switches* podem falhar, um cabo pode ser cortado ou desconectado, mas se a rede tiver uma topologia redundante entre tais *switches*, há uma garantia de disponibilidade do serviço para maioria dos usuários.

Porém, redes de computadores com *links* redundantes trazem a possibilidade dos quadros (dados que trafegam entre os computadores) entrarem em *loop*, causando assim problemas de performance na rede.

Para resolver este problema, redes locais usam o *Spanning Tree Protocol* (STP), que previne que os quadros entrem em *loop* indefinidamente através destes em *links* redundantes.

Sem STP, os dados enviados pelos computadores entrariam em *loop* por um período de tempo indefinido na rede através dos links físicos redundantes. Para evitar o loop de quadros, o STP bloqueia as portas do *switch* de enviar informações garantindo assim somente um caminho ativo entre pares de segmentos de redes locais (domínio de colisão).

O resultado do STP é ao mesmo tempo bom e ruim. Como os quadros não ficarão em *loop*, a rede estará sem problemas de desempenho e usável. No entanto, a rede não tira vantagem dos *links* redundantes ativamente. Isto porque eles são bloqueados para evitar os *loops*. Algumas informações trafegam aparentemente por um caminho mais longo pela rede, porque o eventual menor caminho pode estar bloqueado, o que é ruim. De qualquer forma, o resultado global é bom. Quando quadros trafegam indefinidamente por uma rede, podem torná-la extremamente lenta e impossível de ser usada. Portanto, o STP tem alguns efeitos colaterais em proporções menores comparado ao maior beneficio de permitir a implementação de redes redundantes.

O STP é um protocolo orientado à camada 1 do modelo TCP/IP (Camada de Enlace) desenvolvido originalmente pela DEC (*Digital Equipment Corporation*) e mais tarde incorporado pelo padrão IEEE 802.1d.

Seu objetivo é permitir a comunicação entre os *switches* participantes em uma rede Ethernet, oferecendo a redundância necessária ao mesmo tempo em que evitando a ocorrência de loops na rede. Em redes de camada 2 (Rede), o protocolo de roteamento é o elemento responsável pela convergência de links faltosos, ao mesmo tempo em que estes

protocolos oferecem mecanismos para evitar o *loop* na rede. Estas mesmas necessidades também se fazem presentes em redes onde não há protocolos de roteamento, como é o caso de uma rede composta por *switches* Ethernet, sendo este o tema deste projeto.

De acordo com ODOM (2004), o algoritmo STP cria uma árvore de cobertura (*Spanning Tree*), da qual fazem parte somente as portas dos *switches* que encaminham informações. A estrutura da árvore é formada por um caminho único entre os segmentos que interligam todos os switches da rede. Pode-se entender melhor imaginando um único caminho para se chegar do topo à base de uma árvore.

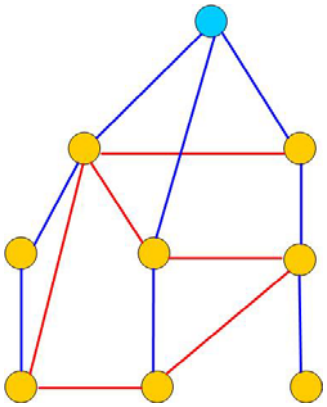


Figura 2.24: Esquema de uma árvore de cobertura.

Fazendo-se das palavras de KUROSE & ROSS (2003), para evitar circulação e multiplicação de quadros, os switches usam o protocolo *Spanning Tree*. Através dele, eles se comunicam pela LAN para determinar uma árvore, isto é, um subconjunto da topologia original que não tenha *loops*. Assim que os *switches* determinam essa árvore, elas desconectam as interfaces apropriadas para criar a árvore como topologia ativa a partir da topologia original. Desconectadas as interfaces e removidos os *loops*, os quadros não vão mais circular e se multiplicar. Se, algum tempo depois, um dos enlaces falhar, os *switches* poderão rodar novamente o algoritmo de *Spanning Tree* e determinar um novo conjunto de interfaces a serem desconectadas.

O *Spanning Tree Protocol* é empregado nos *switches* Ethernet para que estes possam construir uma topologia redundante, porém sem loops. O seu algoritmo é utilizado para o cálculo de todos os caminhos possíveis, permanecendo ativos somente aqueles que forem considerados como os mais eficientes, devendo os links redundantes permanecerem em uma condição "*backup*". Esta é a grande diferença entre a prevenção de *loops* oferecida pelos protocolos de roteamento e o mesmo serviço oferecido pelo *Spanning Tree Protocol*.

Com o STP, não é possível haver loops redundantes ativos, com a exceção de alguns modelos em particular.

2.8.1 Funcionamento

Antes de conhecer o funcionamento do STP, o prévio conhecimento de alguns parâmetros se faz necessário (KUROSE & ROSS, 2003).

- *Bridge Protocol Data Unit* (BPDU): o BPDU é um quadro de *multicast* que os *switches* periodicamente geram para compartilhar informações da topologia e para eleger o *Switch Root* ou Switch Raiz, que vai construir o STP e impedir os *loops*, ativando os *links* quando necessários.

- *Path cost*: cada porta tem associada a ela um custo, que geralmente é o valor inverso da largura de banda da porta. O menor custo é usado quando dois existem para o mesmo destino.

- *Port Priority*: cada porta tem uma prioridade padrão; se dois caminhos que levam a um destinatário existirem e o acumulado do custo por porta for o mesmo, a porta com maior prioridade é preferencial, o menor valor tem maior prioridade. Se duas prioridades forem iguais, aquela com menor numeração física é escolhida.

Segundo ODOM (2004), o algoritmo *Spanning Tree* (árvore de cobertura) coloca cada porta do *switch* no estado de *Forwarding* (FW) ou *Blocking* (BL). Todas as portas no estado FW, ou seja, que podem transmitir dados, fazem parte da atual árvore de cobertura (*Spanning Tree*). O conjunto de portas no estado FW cria um caminho único onde as informações podem ser enviadas entre os segmentos da rede local (LAN). Em suma, *switches* podem encaminhar ou receber dados através das portas que estiverem no estado FW; e não encaminham ou recebem dados através das portas bloqueadas (BL).

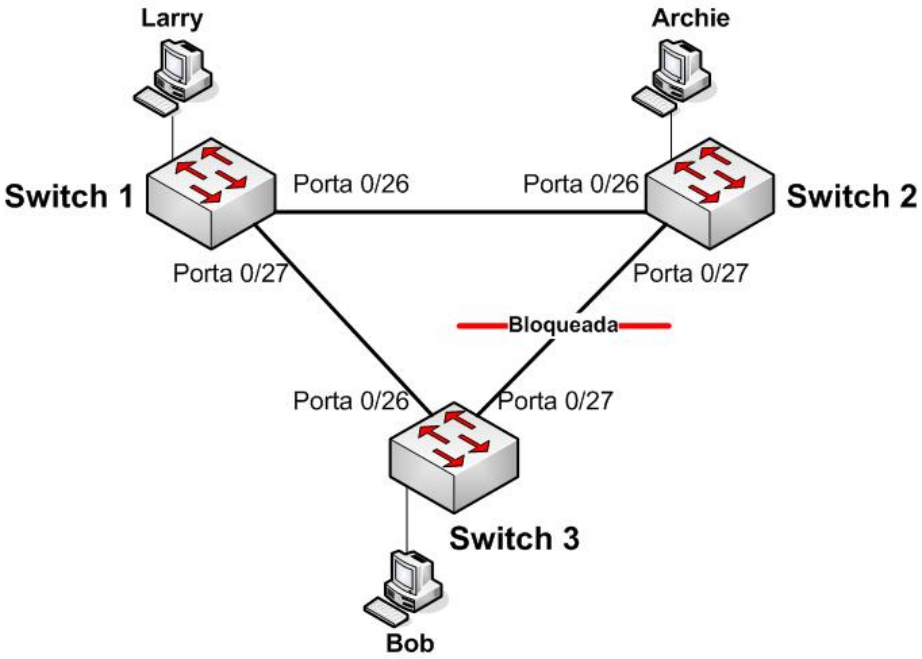


Figura 2.25: Switches com redundância e Spanning Tree Protocol.

Na Figura 2.25, quando *Larry* envia um quadro de *broadcast*, este quadro não entra em *loop*. O Switch1 envia uma cópia para o Switch3, mas este não pode encaminhar o quadro para o Switch2 através de sua porta 27, que está bloqueada pelo STP. O Switch1 envia um *broadcast* para o Switch2, que o encaminha para o Switch3. Esta informação não é recebida, pois o Switch3 ignora informações que entram na porta 27.

Um problema observado nesta topologia é que se *Archie* enviasse um quadro para Bob, esta informação tomaria um caminho mais longo (através de *Larry*), para chegar em *Bob*. Mas com a atual alta tecnologia em redes de computadores, onde as informações trafegam rapidamente a velocidades de 100 Mbps, tal diferença de performance seria praticamente imperceptível, salvo os casos onde a rede esteja realmente comprometida por algum outro problema.

Se o enlace entre o Switch1 e o Switch3 falha, o STP converge, ou seja, reorganiza sua topologia. Assim, o Switch3 não mais bloqueia informações na sua porta 27. A Figura 2.26 mostra como fica a topologia da rede após uma eventual falha no enlace entre os Switches 1 e 3.

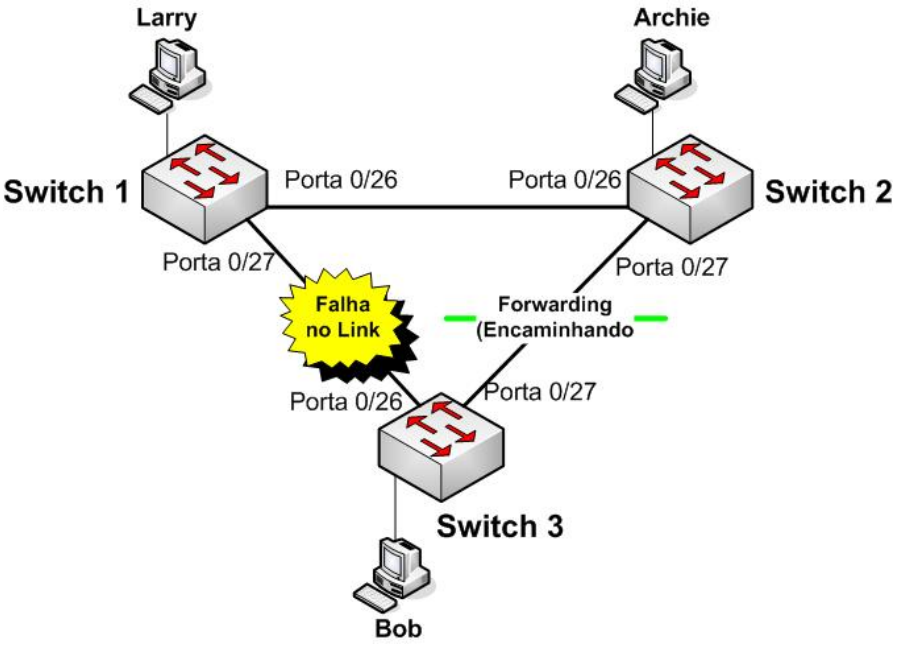


Figura 2.26: Rede com links redundantes após uma falha.

Para decidir quais portas estarão no estado de encaminhamento (FW) e no estado bloqueado (BL), o STP usa três critérios:

1. Eleição de um *Switch* Raiz (*Root*). Todas as portas do *Switch* Raiz (SR) são colocadas no estado FW.

2. Cada *switch* Não-Raiz (*Non-Root*) considerada uma de suas portas como tendo o menor custo entre ele e o SR. O STP coloca então esta porta de menor custo (*Least Root Cost Interface*), chamada *Switch Root Port*, ou simplesmente Porta raiz (PR), no estado FW.

3. Muitos *switches* podem fazer parte de um mesmo segmento de rede local. O *switch* com o menor custo até SR, quando comparado com os outros *switches* deste segmento, é colocado também no estado FW. O switch com o menor custo em cada segmento é chamado de *Switch* Designado (SD), e a porta deste switch, conectada a este segmento, é chamada de *Designed Port* ou Porta Designada (PD).

Tabela 2.2: Razões para o STP Encaminhar ou Bloquear.

Caracterização da Porta	Estado STP	Descrição
Todas as portas do *Switch* Raiz	FW	O SR é sempre o *Switch* Designado em todos os segmentos conectados.
Cada porta do *Switch* Não-Raiz	FW	A Porta Raiz é a porta que recebe o BPDU de menor custo do SR.
Cada Porta Designada da rede	FW	O switch que encaminha o BPDU de menor custo no segmento é o *Switch* Designado para este segmento.
Todas outras portas	BL	A porta não é usada para encaminhar dados, e também não recebe informações das portas no estado FW.

Elegendo o Switch Root e descobrindo as Portas Raiz e Portas Designadas (PD)

O STP começa com cada *switch* enviando mensagens pedindo para ser o *Switch Root*. O algoritmo define estas mensagens, para troca de informações com outros switches, como *Bridge Protocol Data Units* (BPDU). Cada *switch* começa enviando um BPDU especificando:

- O Identificador (ID) do SR: O *Switch ID* é composto pela concatenação do *Switch Priority* (prioridade do *switch*) com seu endereço *MAC*. Quanto menor o valor do *Switch Priority*, maior a chance de se tornar o SR. A especificação do padrão IEEE 802.11d permite prioridades entre 0 e 65.535.

- O custo para alcançar o SR a partir de cada *switch*: No começo do processo, quando cada switch requisita o estado de SR, todos *switches* têm custo 0, que é equivalente ao custo para alcançar ele mesmo. Melhores caminhos apresentam menores custos.

- O *Switch ID* do remetente do BPDU: Este valor é sempre o *Switch ID* do remetente do BPDU, sem levar em consideração se o *switch* que está enviando o BPDU é o SR.

Os *switches* elegem o SR com base nos Switch ID dos BPDUs. O SR é aquele com menor valor numérico de Switch ID. Por exemplo: Se um *switch* tem prioridade 100 e outro *switch* participante da topologia STP tem prioridade 200, aquele com prioridade 100 vence, não precisando comparar o endereço MAC, que compõe a outra parte do *Switch ID*.

Quando um empate ocorre entre as prioridades, o endereço MAC é usado para definir o vencedor. Por exemplo: Um *switch* apresenta prioridade 100 e endereço MAC 0020.0000.0000. Outro *switch* que também tem prioridade 100, porém um endereço MAC 0FFF.FFFF.FFFF, perderia a eleição de SR para o primeiro, que apresenta menor valor numérico de endereço MAC.

A mensagem usada para identificar o SR, seu ID, e custo é chamada *Hello BPDU*.

O *Spanning Tree Protocol* elege o SR de uma maneira não muito diferente de uma eleição política. O processo de escolha do SR começa com todos *switches* enviando BPDUs, com sua ID e prioridade, pedindo para ser o SR. Se um *switch* descobre um candidato melhor, pára de fazer propaganda de si mesmo e começa encaminhar as mensagens (*Hello*) enviadas pelo melhor *switch*. O candidato inferior apóia outro candidato melhor colocado.

A Figura 2.27 ilustra parte do processo. Imagine que o Switch1 esteja se promovendo para ser o Raiz, assim como o Switch2 e Switch3.

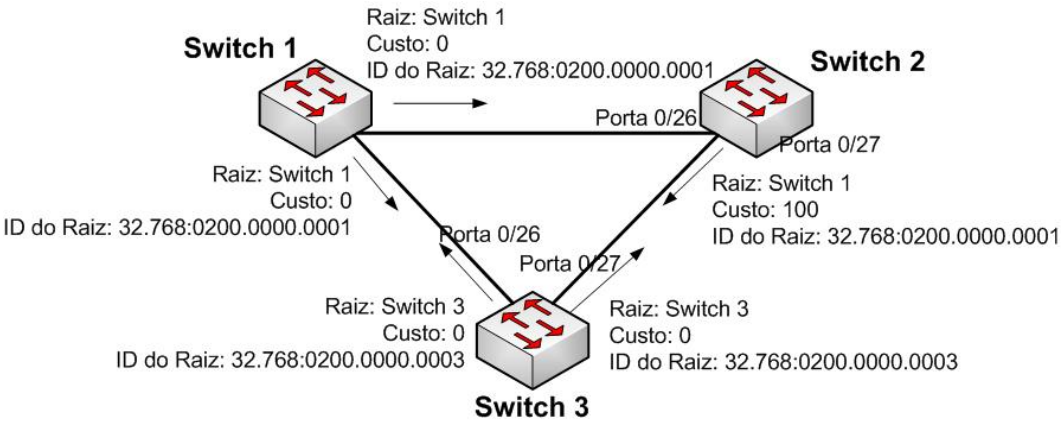


Figura 2.27: Processo de eleição do SR.

O Switch2 sabe que o Switch1 é melhor, mas os Switch1 e 3 ainda acreditam ser os melhores, então eles ainda promovem-se como Raiz.

Dois candidatos ainda existem. Portanto, para definir o vencedor, aquele com menor prioridade vence. Caso haja empate, aquele com menor endereço MAC torna-se o SR.

Como mostra a Figura 2.28, o Switch1, como menor ID, vence. Também mostra o resultado da troca de mensagens de configuração (*Hello Messages*) entre os *switches*.

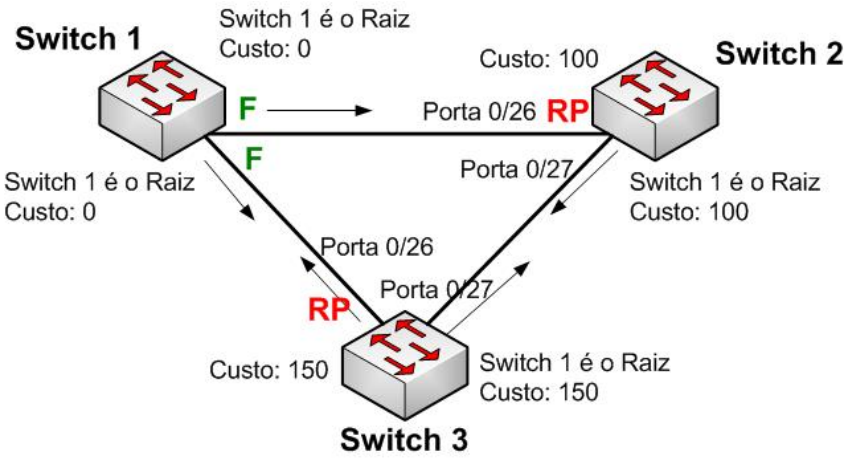


Figura 2.28: Estados das Portas quando o switch 1 vence a eleição.

Todas as portas do Switch1 são colocadas no estado FW (ativo), porque o Switch1 venceu a eleição de SR. A segunda razão que faz o STP colocar uma porta para encaminhar quadros é se a porta naquele *switch* é a Porta Raiz. Cada *switch* tem uma única PR, que é a porta que recebe o BPDU de menor custo do SR.

Na Figura 2.27, o Switch2 tem o melhor custo nas mensagens de configuração entrando na porta 0/26. Da mesma forma, o melhor custo que está entrando no Switch3 usa a porta 0/26. Então, o STP configura estas portas como PR e as coloca no estado FW.

O custo é calculado adicionando o custo recebido na mensagem de configuração (0 neste caso) ao custo da porta onde a mensagem foi recebida. Logo, o Switch2 soma 100 a 0, e o Switch3 adiciona 150 a 0 (Figura 2.25).

A porta 0/27 do Switch2 é a Porta Designada no segmento entre o Switch2 e Switch3. Isto porque o Switch2 envia o BPDU de menor custo (Switch1).

No exemplo da Figura 2.28, a única porta dos três *switches* que não encaminha dados é a porta 0/27 do Switch3. Assim o processo se completa, com todas as portas no estado FW exceto a 0/27 do Switch3, que permanece no estado BL, até que a topologia mude.

O custo das portas pode ser configurado, ou pode-se usar os valores padrões. A Figura 2.29 mostra os custos padrões, definidos pelo IEEE, os mesmos usados neste projeto. Os valores originais foram apresentados nos anos 80 e não previam o crescimento do padrão Ethernet para suportar as redes Gigabit. Houve então uma revisão destes custos.

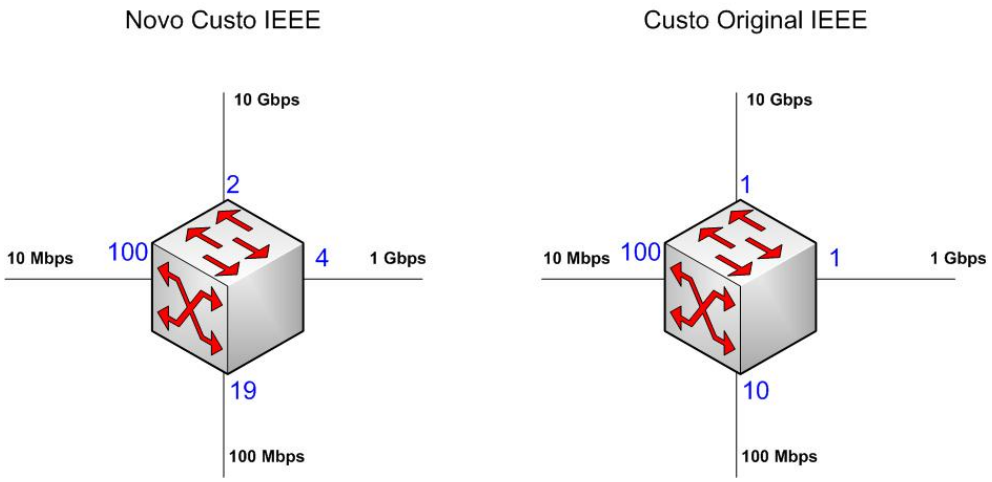


Figura 2.29: Custos padrões das portas (em azul), de acordo com o IEEE.

Campos do quadro BPDU

***Protocol Identifier*:** Identificador do Protocolo. Contém o valor 0.

***Version*:** Versão. Contém o valor 0.

***Message Type*:** Tipo de Mensagem. Contém o valor 0.

Flags: Este campo pode conter um entre dois eventos, são eles: alterações na topologia ou aceitação (ACK) na modificação da topologia.

***Root Identifier*:** Identificador Raiz. Define o *switch* sendo o de mais alto nível no STP.

***Port Identifier*:** Identificador da porta. Define por que porta o BPDU sai do *switch*. Isso é usado por outros switches para detectar e remover *loops* na rede.

***Message Age*:** Tempo da Mensagem. Define o tempo final em que o *switch* Raiz publica o BPDU, baseado no estado atual de configuração da rede.

***Maximum Age*:** Tempo máximo. Define a tempo em que o protocolo vai remover a informação do banco de dados e iniciar a alteração na topologia através da execução do algoritmo *Spanning Tree*. Este parâmetro é importante, pois permite que os *switches* rodem o algoritmo de forma uniforme e em paralelo.

***Hello Time*:** Tempo de Alô. Intervalo em que o *switch* publica o BPDU.

Forwarding Delay: Atraso de encaminhamento. Tempo em que a porta vai permanecer em um determinado estado.

Como os *switches* compartilham os BPDUs, será descoberta qual é a estrutura atual da topologia, incluindo os identificadores de cada switch.

2.8.2 Reagindo a mudanças na Rede

Depois que a topologia STP é montada, ela não muda a menos que a topologia da rede mude, ou seja, um enlace caia, um *switch* trave, entre outras razões. O SR envia *Hello* BPDUs a cada 2 segundos, por padrão. Cada *switch* repassa este *Hello*, adicionando seu custo ao custo para alcançar o SR. Cada *switch* sempre "escuta" as mensagens do SR como um meio de saber se seu caminho até o Raiz está funcionando, visto que as mensagens de configuração seguem o mesmo caminho dos quadros de dados. Quando um *switch* pára de receber estas mensagens, algum caminho falhou, então ele reage e inicia o processo de mudança de topologia do *Spanning Tree*.

A mensagem de configuração (*Hello* BPDU) define os tempos adotados por todos os *Switches*.

- *Hello Time* (Tempo de *Hello*): quanto tempo o SR espera antes de enviar as periódicas mensagens de configuração (*Hello* BPDU), que são repassadas sucessivamente entre os switches da rede. O padrão é 2 segundos.

- *Max Age* (Tempo Máximo): Quanto tempo um *switch* deve esperar, depois de não mais receber mensagens de configuração, antes de mudar a topologia STP. Geralmente é adotado um valor múltiplo da mensagem de configuração; o padrão é 20 segundos.

- *Forward Delay* (Atraso de Encaminhamento): Atraso que afeta o tempo envolvido enquanto uma porta muda do estado BL para o estado FW. Uma porta fica no estado de *Listening* (LN), ou "escutando" durante o tempo definido pelo *Forward Delay*.

A Figura 2.30 mostra uma falha no *link* entre o Switch1 e Switch3:

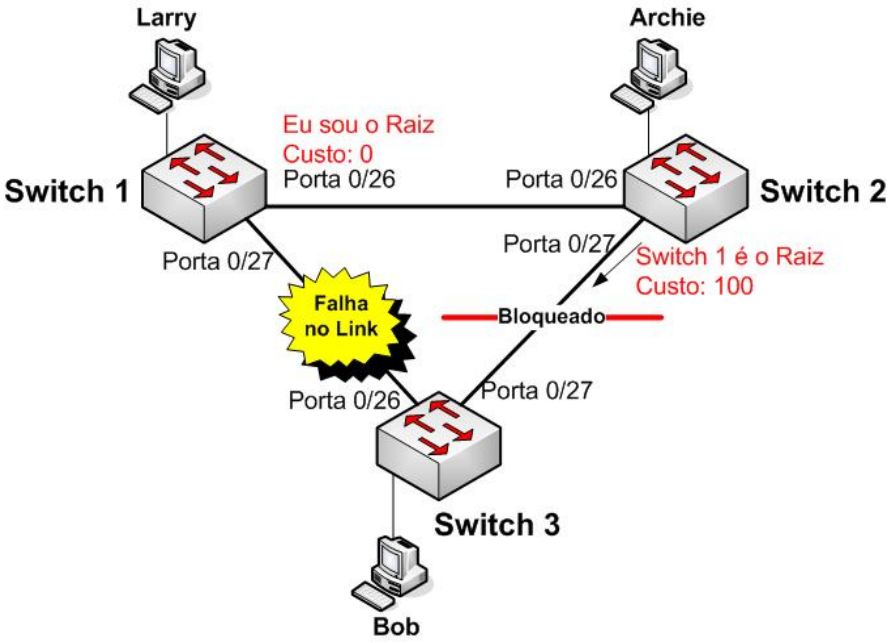


Figura 2.30: Convergência do STP.

O Switch3 reage à mudança, mas o Switch2 não. O Switch3 pára de receber mensagens de configuração em sua Porta Raiz (0/26). Lembrando que quando um *switch* pára de ouvir as melhores mensagens de configurações após o período de *Max Age* (tempo máximo), ele reage. No entanto, o Switch2 continua recebendo os BPDUs, então não reage.

Depois que o tempo de máximo expira no Switch3, ele ou promove-se como SR, ou reconhece como SR aquele que é o Raiz. Como Switch2 encaminha o pedido do Switch1 para ser o Switch Raiz e o Switch 1 já é o Raiz, Swicth1 deve ter uma melhor prioridade ou melhor endereço MAC que o Switch 3. Em outras palavras, neste caso, o Switch3 já sabe que perde a eleição de SR para o Switch1. Então o Swtich3:

- decide tornar sua porta 0/27 a Porta Raiz, pois está recebendo mensagens de configuração com menor *Switch ID* através desta porta. Então o Switch3 a coloca no estado FW;
- a porta 0/26 provavelmente teve uma falha física, então é colocada no estado BL;
- o Switch3 limpa sua tabela de endereços para estas duas portas porque a localização dos endereços MAC, relativos a ele mesmo, pode ter mudado. Por exemplo, o endereço MAC de *Larry* era anteriormente alcançado através da porta 0/26 e agora é alcançado pela porta 0/27.

Todavia, o Switch3 não pode mudar imediatamente o estado da porta 0/27 de BL para o estado FW. Se o Switch3 mudar instantaneamente seu estado para FW, e algum outro *switch* também estiver convergindo, *loops* podem ocorrer na rede, o que é indesejável. Para prevenir este problema, STP usa dois estados intermediários das portas.

O primeiro, *Listenning State* (LN), "ouvindo", permite a cada dispositivo esperar para ter certeza que não já melhores mensagens de configuração, indicando o SR. O segundo, *Learning State* (LR), "aprendendo", permite ao *switch* identificar a nova localização dos endereços MAC sem entrar no estado FW e possivelmente causar *loops*. Estes estados previnem que os *switches* inundem a rede com quadros até que todos os outros *switches* tenham convergido e aprendido os novos endereços MAC.

Usando os tempos padrões, o Switch3 precisa de 50 segundos até colocar uma porta *Fast Ethernet* no estado FW (ativo), ou seja, o tempo que leva para convergir. Primeiro, o *switch* que falhou espera 20 segundos (*Max Age Time*) antes de decidir que ele não está mais recebendo os mesmos BPDUs em sua porta Raiz. Neste ponto, o Swtich3 coloca a porta 0/27 no estado LN por 15 segundos (*Forward Delay*, de acordo com o padrão). Depois, coloca a porta 0/27 no estado LR também por 15 segundos (*Forward Delay*), antes de mudar o estado da porta para o estado FW (*Forwarding State*). A Tabela 2.3 mostra o resumo dos estados do STP.

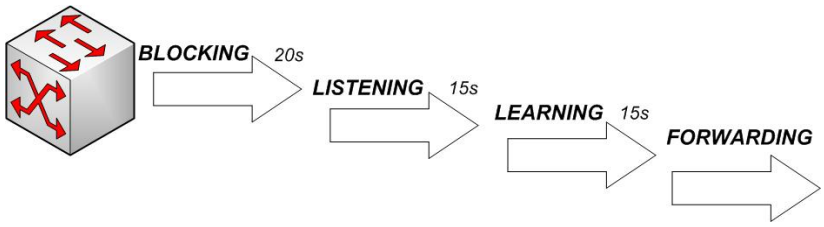


Figura 2.31: Passagem de estados.

Tabela 2.3: Resumo dos estados do STP.

Estado	Encaminha quadros?	Aprende endereços MAC ?	Estado transitório ou estável?
BL	Não	Não	Estável
LN	Não	Não	Transitório
LR	Não	Sim	Transitório
FW	Sim	Sim	Estável

O Switch3 também deve avisar aos outros switches para atualizar as entradas em suas tabelas. Por exemplo, a tabela do Switch2 lista o endereço MAC de Bob sendo encaminhado pela porta 0/26, mas agora ele será feito através da porta 0/27. Então, o Switch3 envia uma mensagem de configuração especial, chamada de TCN (*Topology Change Notification* – Notificação de Mudança de Topologia) BPDU através da porta 0/27. Quando o Switch2 recebe a notificação de mudança de topologia (TCN), ele suspende temporariamente todas entradas MAC em sua tabela. Pelo fato do Switch3 enviar o TCN BPDU tão logo quanto ele coloca suas portas no estado LN, o Switch2 removeu a entrada para o endereço MAC de Bob antes que o Switch3 comece a encaminhar quadros. O Switch2 também repassa o TCN até o SR, que certifica-se que todas os outros *switches* tenham atualizado suas entradas na tabela de endereços MAC.

2.8.3 Características Opcionais do STP

Segundo ODOM (2004), o STP foi desenvolvido há cerca de 20 anos e concebido para resolver um problema bastante especifico nas redes, que têm mudado ao longo destes anos. Da mesma forma, fornecedores e padrões têm trazido constantemente melhorias ao STP. A Cisco, empresa de comunicações que fabrica *switches* com suporte à tecnologias como STP, assim como o IEEE, que detêm as especificações do STP, desenvolveram novas tecnologias que têm melhorado consideravelmente o desempenho do STP original: o EtherChannel e o Port Fast.

EtherChannel

Ainda segundo ODOM (2004), a melhor maneira de diminuir o tempo de convergência padrão do STP, que é muito elevado (50 segundos), é simplesmente evitando a convergência. A tecnologia Etherchannel provê uma maneira de evitar a necessidade de convergência do STP quando uma falha em um cabo ou uma porta ocorre.

Esta tecnologia usa uma combinação de 2 a 8 troncos (*trunk*) Ethernet paralelos entre o mesmo par de *switches*, encapsulados em um Etherchannel. O STP trata o EtherChannel como um enlace único, então se apenas um dos caminhos estiver ativo, a convergência STP não tem que acontecer.

A Figura 2.32 mostra uma rede com 3 *switches* e 2 ligações entre cada par de *switch*.

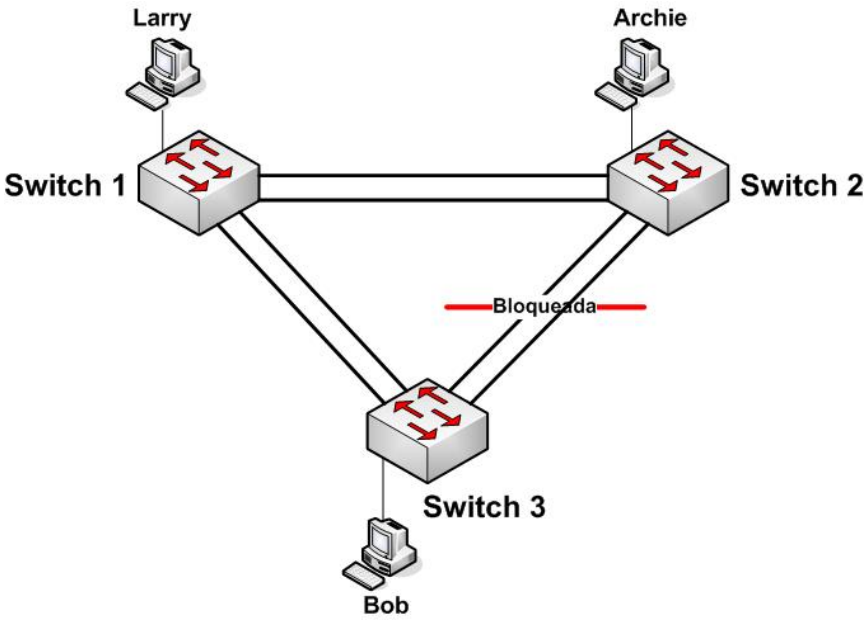


Figura 2.32: Etherchannel com 2 troncos entre os switches.

Na Figura 2.32, o STP convergiria somente se os dois *links* entre um par de *switches* falhassem ao mesmo tempo, o que possível, mas improvável. Sem o Etherchannel, caso haja múltiplos links paralelos entre dois *switches*, STP bloqueia todos com exceção de um. Usando Etherchannel, todos os *links* paralelos podem estar ativos e funcionando ao mesmo tempo, reduzindo assim o número de vezes que o STP converge, o que aumenta a disponibilidade da rede.

Este recurso também provê um aumento de banda disponível na rede. Todos os canais Etherchannel estão em estado FW ou BL, pelo fato do STP tratar todos os troncos no mesmo Etherchannel como um único tronco. Quando um Etherchannel está no modo FW, os *switches* encaminham trafego através de todos os troncos, disponibilizando uma maior largura de banda.

PortFast

A avanço previsto pelo PortFast permite ao *switch* colocar uma porta no estado FW imediatamente quando a porta torna-se fisicamente ativa. Em outras palavras, logo quando um computador é ligado na rede, o *switch* já o coloca no estado FW (a porta a qual o *switch* foi ligado).

Se o PortFast estiver ativo na porta em que um usuário acaba de ligar seu computador e entra na rede, este computador imediatamente pode encaminhar dados. Sem o PortFast, cada porta deve aguardar 50 segundos (Tempo de *Max Age*, 20 segundos, somado a duas vezes o *Forward Delay*, 15 segundos).

2.9 *Rapid Spanning Tree Protocol* (IEEE 802.1w)

Como mencionado na seção anterior, o IEEE define o STP no padrão 802.1d. O IEEE desenvolveu novas características a fim de melhorar a performance do STP original, denominando o novo padrão de *Rapid Spanning Tree Protocol* (RSTP), que é definido pelo padrão 802.1w.

ODOM (2004) explica que o RSTP trabalha semelhantemente ao STP (802.1d) em vários aspectos, como:

- elege o *Switch Root* (SR) usando os mesmos parâmetros e o mesmo critério de desempate;
- elege as Portas Raiz (PR) nos *switches* que não são o SR seguindo as mesmas regras;
- elege as Portas Designadas (PD) em cada segmento LAN da mesma maneira;
- coloca cada porta, seja no estado de *forwarding* (FW) ou bloqueado (BL) – apesar do RSTP chamar o *Blocking State* (BL) de *Discarding State* (DC).

O RSTP pode ser utilizado paralelamente em *switches* com o STP original, com as características RSTP rodando naqueles que têm suporte a ele, e as características STP funcionando nos switches com suporte somente ao padrão 802.1d.

Apesar de todas as similaridades citadas entre STP e RSTP, a principal razão que levou o IEEE a desenvolver o novo padrão foi a convergência. O STP leva um tempo relativamente longo para convergir (50 segundos, usando as configurações padrões). O RSTP melhora a convergência da rede quando uma mudança na topologia ocorre.

Os três períodos padrões de espera do STP – *Max Age*, 20 segundos, somado às duas transições entre os estados intermediários, *Forwarding Delay*, 15 segundos – geram uma convergência muito lenta do STP.

O tempo de convergência do RSTP leva tipicamente menos que 10 segundos. Em alguns casos pode ser inferior, como 1 ou 2 segundos.

Para melhor compreender o funcionamento do RSTP, alguns conceitos e terminologias precisam ser conhecidos, como os de conexões físicas.

O RSTP caracteriza as conexões físicas em uma rede em três diferentes tipos: 1) *Link-type point-to-point*; 2) *Link-type shared*; 3) *Edge-type*.

A Figura 2.33 mostra cada tipo:

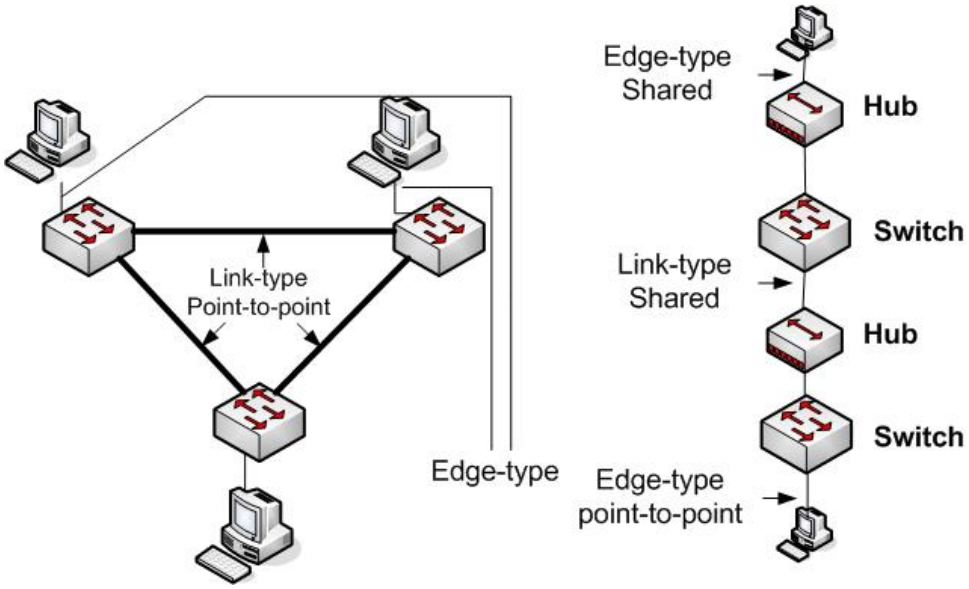


Figura 2.33: Tipos de enlaces no RSTP.

No exemplo da Figura 2.33, a rede da esquerda trabalha somente com *switches*, enquanto a outra faz uso de *hub*, além de *switches*.

O IEEE não prevê o funcionamento do RSTP em redes onde há *hubs* interconectando as estações, não apresentando seu efetivo tempo de convergência nessas redes.

O RSTP denomina os enlaces entre *switches* de *links* e chama as ligações entre uma estação e o equipamento de interconexão (*hub* ou *switch*) de *edges.*

RSTP reduz o tempo de convergência somente para as ligações entre dois *switches* (*link-type point-to-point*) e entre uma estação e um *switch* (*edge-type*). Para uma ligação entre um *hub* e um *switch* ele não produz resultados satisfatórios.

2.9.1 Estados das Portas no RSTP

A Tabela 2.4 mostra um comparativo entre os estados do RSTP em relação ao STP:

Tabela 2.4: Estados das portas no RSTP e STP.[5]

Estado operacional	Estado STP	Estado RSTP	Porta faz parte da topologia ativa?
Ativada	*Blocking* (BL)	*Discarding* (DC)	Não
Ativada	*Listening* (LN)	*Discarding* (DC)	Não
Ativada	*Learning* (LR)	*Learning* (LR)	Sim
Ativada	*Forwarding* (FW)	*Forwarding* (FW)	Sim
Desativada	Desativada	*Discarding* (DC)	Não

Similarmente ao STP, o RSTP estabiliza todas suas portas, ou no estado FW, ou no DC. DC, ou *Discarding*, significa que a porta não envia nem recebe pacotes, e também não aprende endereços MAC. Ela somente ouve os BPDUs.

Em resumo, o DC se comporta como o estado BL do STP. RSTP usa um estado intermediário (LR), que funciona da mesma forma que no STP. A diferença é que no RSTP o LR é usado por um curto período de tempo (ODOM, 2004).

2.9.2 Tipos de Portas no RSTP

O *Rapid* STP define três novas portas de um *switch*. A Figura 2.34 mostra dois deles. Ainda existe um terceiro tipo (*Disabled*), que não é visto nesta figura. As portas incorporadas ao RSTP são:

- *Backup Port* ou Porta Backup (PB);
- *Alternate Port* ou Porta Alternativa (PA);
- Disabled Port ou Porta Desabilitada (PDE);

[5] Fonte: ODOM (2004).

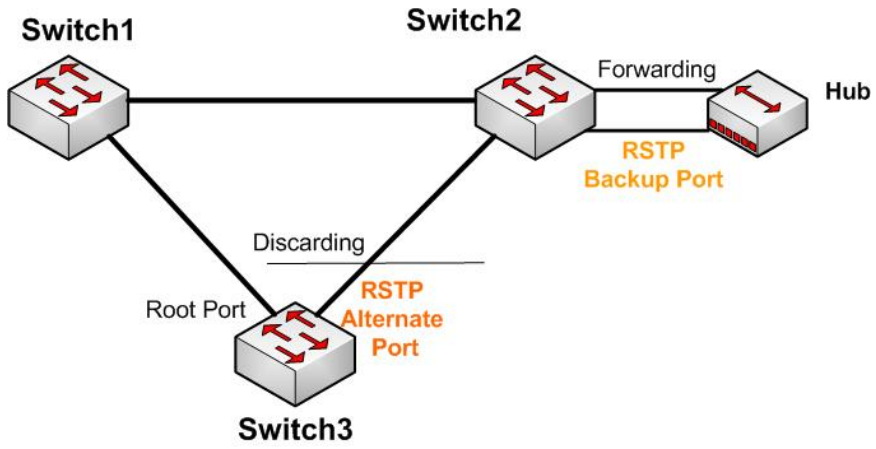


Figura 2.34: Novos tipos de portas do RSTP.

As principais diferenças entre o STP e o RSTP, segundo ODOM (2004), são:

- o RSTP determina que as Portas Alternativas (PA) irão receber os BPDUs que não são tão "bons" quanto os recebidos pela Porta Raiz. Assim, se um *switch* pára de receber BPDUs do Switch Raiz, RSTP escolhe a melhor PA como nova Porta Raiz, acelerando assim o processo de convergência.
- a outra porta nova (PB) é usada somente quando um único *switch* tem dois enlaces para o mesmo segmento. Para que isso aconteça o *switch* deve estar conectado a um hub, como mostra a Figura 2.34. O princípio de funcionamento é o mesmo: O *switch* coloca uma das portas no estado FW e a outra no DC. Com isso o *switch* envia BPDUs pela porta FW e recebe os mesmos na porta DC. Então ele sabe que tem uma conexão extra para aquele segmento, assim chamada de *Backup Port* (BP). Caso a porta que está no estado FW falhe, o RSTP no *switch* coloca imediatamente a BP no estado FW.

A possibilidade de colocar uma porta que anteriormente estava bloqueada (BP ou PA) imediatamente no estado FW, sem ter que passar pelos estados intermediários LN e LR, é a razão do menor tempo levado pelo RSTP para convergir.

3 MATERIAIS E MÉTODOS

3.1 Tipos de Pesquisa

Neste trabalho foram realizadas as pesquisas bibliográfica, documental e a pesquisa-ação. A pesquisa bibliográfica serviu como base para a aquisição de conhecimento acerca dos temas envolvidos no projeto, como, por exemplo, funcionamento dos diversos protocolos de rede e as especificações técnicas do *Spanning Tree Protocol* (STP). Envolveu, basicamente, consultas a livros de referência, teses científicas e artigos da área de tecnologia de redes de computadores.

Pesquisa-ação é um tipo de pesquisa social com base empírica que é concebida e realizada em estreita associação com uma ação ou com a resolução de um problema coletivo, no qual os pesquisadores e os participantes representativos da situação ou problema estão envolvidos de modo cooperativo ou participativo (THIOLLENT, 1997).

A pesquisa documental é constituída pelo exame de materiais que ainda não receberam um tratamento analítico ou que podem ser reexaminados com vistas a uma interpretação nova ou complementar. Portanto, a pesquisa documental é o levantamento de dados através de estudo minucioso em documentos de fontes primárias encontrados em arquivos públicos, arquivos particulares, fontes estatísticas, fontes não escritas, de forma sistemática com fim de saber um campo qualquer do conhecimento (TRIVIÑOS, 1987).

A pesquisa-ação deste trabalho consistiu em testar uma solução para o problema de disponibilidade e desempenho da Rede Ufla, enquanto a pesquisa documental foi a análise da documentação da rede, equipamentos utilizados, problemas que geralmente ocorrem, bem como a reunião com os administradores da rede para coletar informações.

Nas seções seguintes, serão apresentadas as análises que apontam as principais causas desse problema e também todos os procedimentos realizados para testar soluções que visam minimizá-los.

3.2 Procedimento Metodológico

Realizou-se inicialmente uma pesquisa bibliográfica dos assuntos citados no Capítulo 2 para adquirir um embasamento teórico. Foi necessário conhecer todo o campo das Redes de Computadores – infra-estrutura de redes, arquitetura de protocolos (TCP/IP)

e especificamente o funcionamento do *Spanning Tree Protocol* (Protocolo de Árvore de Cobertura). Além disso, foi necessário conhecer amplamente o funcionamento de alguns equipamentos, principalmente dos *switches*, que são dispositivos indispensáveis em uma arquitetura de rede do tipo Campus, como é o caso da Rede Ufla.

Uma questão importante para iniciar este trabalho foi tomar conhecimento dos problemas enfrentados freqüentemente pelos administradores da Rede Ufla, bem como as necessidades de seus usuários. Fez-se necessário, portanto, um estudo bastante aprofundado sobre o assunto.

Foi feito também, uma análise das derivações do *Spanning Tree Protocol*, para verificar qual melhor se adaptaria às especificidades da Rede Ufla. Devido às restrições tecnológicas da Rede Ufla, apenas o STP original pôde ser testado e, provavelmente, somente este também poderá ser implantado.

Após a configuração e execução do STP, foi também realizado um monitoramento do tráfego da rede simulada e, finalmente, a análise dos resultados.

Resumindo o exposto acima, para o desenvolvimento deste projeto, foram realizadas as seguintes atividades:

1. Estudo bibliográfico;
2. Análise das características da rede Ufla e necessidades de otimização;
3. Configuração do *Spanning Tree:* A configuração do STP na etapa de testes foi feita via *software* de gerenciamento (acessado pelo navegador) dos *switches;*
4. Monitoramento da atividade e tráfego da rede antes e depois de configurado o STP.
5. Escrita da Monografia

3.3 Desenvolvimento

3.3.1 Análise da Rede Ufla

Após a coleta de informações relevantes para o embasamento teórico, iniciou-se a fase de análise da Rede Ufla, com o intuito de conhecer o ambiente. Nesta etapa foi estudado o Campus da Ufla, sua topologia e hierarquia de rede, equipamentos, serviços disponibilizados, processo de gerência, suporte e pessoal técnico, etc.

A atual disposição física (topologia) da rede da Universidade Federal de Lavras não apresenta caminhos redundantes entre os departamentos. Sendo assim, não há alternativas de tráfego que poderiam oferecer um balanceamento e aumento de disponibilidade da rede. Isso porque, como visto na seção 2.6, a criação da redundância consiste em adicionar mais de uma conexão entre dois pontos distintos, ou seja, prover caminhos alternativos na rede para a transmissão de dados, garantindo a finalização da transmissão mesmo que parte da rede esteja indisponível.

Esta análise contribuiu para a identificação dos principais problemas, equipamentos existentes, limitações e recursos disponíveis. A reunião e discussão da atual situação da rede Ufla com funcionários e pessoas ligadas à área de rede também contribuiu para um melhor conhecimento do ambiente em questão.

O Campus da Universidade Federal de Lavras possui uma área física de 600 ha com uma área construída de 158.359 m². Atualmente dispõe de 16 departamentos (Tabela 3.1), além de laboratórios e prédios administrativos. Está dividido em Campus Histórico e Campus Novo.

Tabela 3.1: Nomes e siglas dos setores da Universidade.

NOME DO SETOR	SIGLA
Campus Novo	
- Reitoria	**-**
- Departamento de Administração e Economia	**DAE**
- Departamento de Agricultura	**DAG**
- Departamento de Biologia	**DBI**
- Departamento de Ciência da Computação	**DCC**
- Departamento de Ciência dos Alimentos	**DCA**
- Departamento de Ciências Exatas	**DEX**
- Departamento de Ciências Florestais	**DCF**
- Departamento de Ciência do Solo	**DCS**
- Departamento de Engenharia	**DEG**
- Departamento de Entomologia	**DEN**
- Departamento de Fitopatologia	**DFP**
- Departamento de Medicina Veterinária	**DMV**

- Departamento de Química	**DQI**
- Departamento de Zootecnia	**DZO**
- Biblioteca	**-**
- Prefeitura do Campus, Almoxarifado	**-**
- Cantina	**-**
- Postos Bancários, Central de Fotocópias, Agência de Correio, Livraria, DCE	**-**
Campus Histórico	
- Centro Assistencial, Odontológico e Serviço Social	**-**
- Alojamentos	**-**
- Museu "Bi Moreira"	**-**
- Restaurante Universitário	**RU**
- Departamento de Educação Física	**DEF**
- Hidráulica (DEG)	**-**
- Creche	**-**
- Hotel Alvorada	**-**
- Fundação de Apoio ao Ensino Pesquisa e Extensão	**FAEPE**
- Gráfica e Editora UFLA	**-**
- Centro de Tecnologia em Informática	**UFLATEC**
- Cooperativa de Consumo	**COPEUFLA**
- Rádio FM Universitária e TV Universitária	**-**
- Laboratório de Idiomas	**-**

A Rede Ufla é uma Rede Campus, definida por JACK (2003) como uma área com grupos de edifícios interconectados através de várias redes locais. O padrão de rede utilizado na Ufla é o Ethernet.

O acesso à internet é feito através das concessionárias de telecomunicações Embratel e Telemar. Os *links* chegam à sala de equipamentos do Centro de Informática, localizado no prédio da Reitoria e, a partir daí, são transmitidos por meio de cabeamento óptico (fibra óptica) a todos os departamentos da Universidade, como mostra a Figura 3.1.

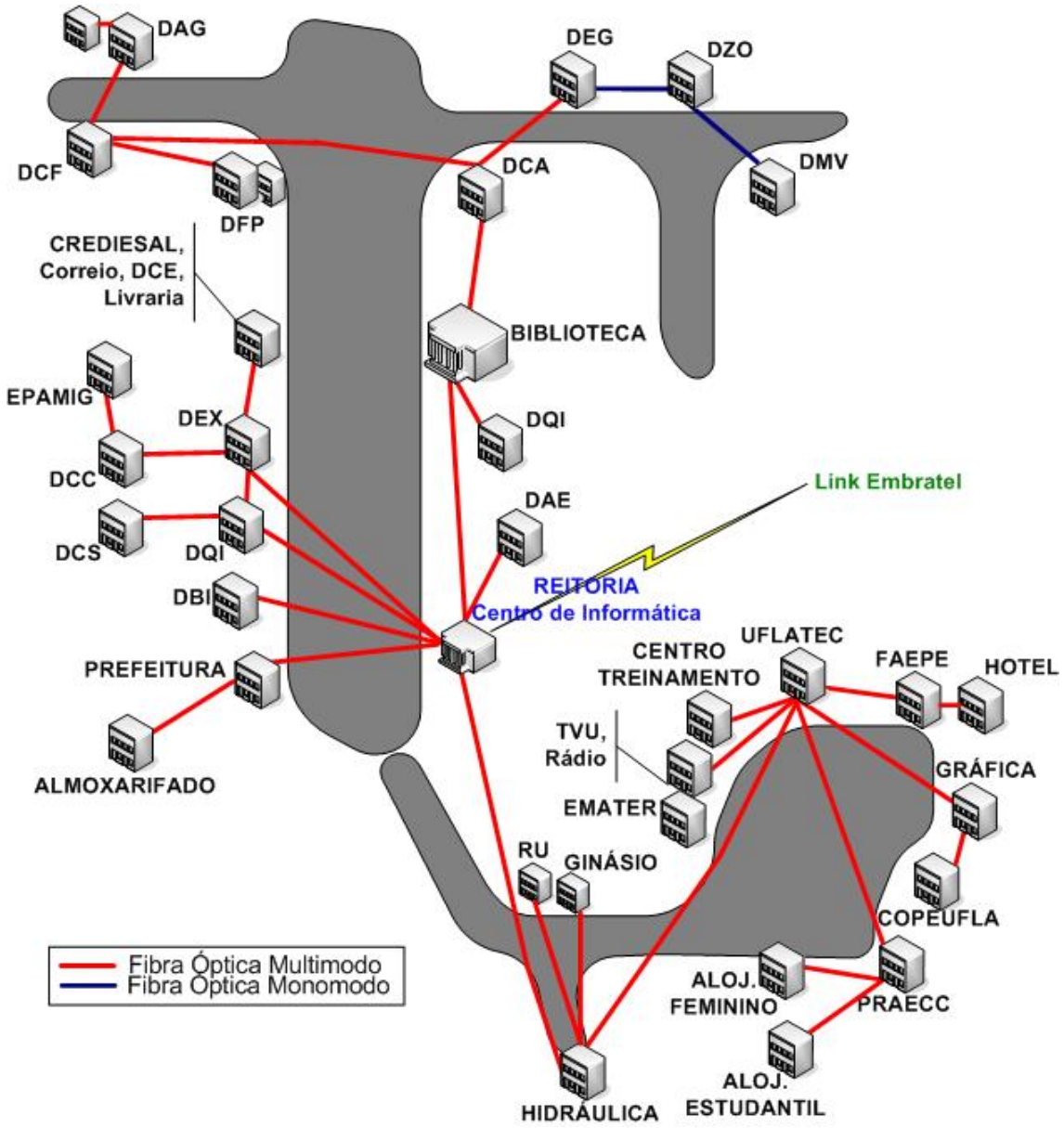


Figura 3.1: Topologia da Rede Ufla[6].

No Centro de Informática (CIN-Ufla) estão localizados os equipamentos que servem de convergência para os *links* departamentais, como o *switch* gerenciável PLANET WGSW-1602, modelo utilizado neste projeto.

[6] Fonte: FRINHANI, R. M. D., Projeto de Reestruturação do Gerenciamento e Otimização da Rede Computacional da Universidade Federal de Lavras. 2004.

Nos principais departamentos da Ufla existe um switch gerenciável (PLANET WGSW 1602) que tem a função de receber o *link* da Rede Ufla e distribuir para os demais equipamentos de interconexão localizados nos outros prédios do próprio departamento, como mostra a Figura 3.2. A interligação destes equipamentos de interconexão com o *switch* gerenciável é feita através de fibra óptica, grande parte fibra óptica monomodo e alguns pontos por fibra óptica multímodo, ambas de 62/125 µm. Fibras monomodo são fibras que possuem um único modo de propagação, ou seja, os raios de luz percorrem o interior da fibra por um só caminho. Já nas multimodo há vários modos de propagação, ou seja, os raios de luz podem percorrer seu interior por diversos caminhos. Umas das deficiências da fibra óptica multimodo é que a banda passante é bastante estreita, restringindo sua capacidade de transmissão. As monomodo, ao contrário, possuem maior largura de banda e atingem maiores distancias, porém têm custo mais elevado.

Um conversor de mídia recebe o sinal óptico da fibra e o converte para sinal elétrico. Este, então, é transmitido para o cabeamento metálico que é conectado ao *switch*. Além dos *switches*, cada departamento utiliza outros equipamentos de rede, principalmente *hubs*. Uma outra função dos *switches* gerenciáveis é a repetição de sinais para os equipamentos de outros departamentos (cascateamento).

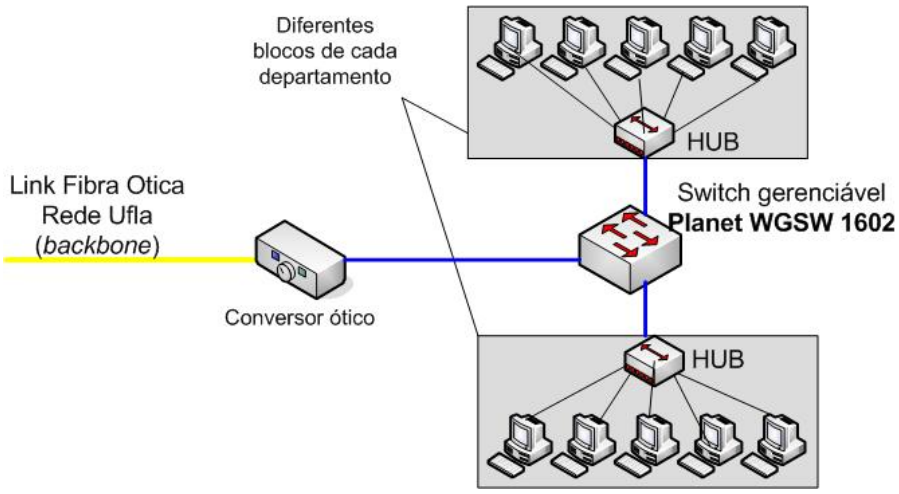


Figura 3.2: Esquema básico da rede de um departamento.

O fluxo de dados na Rede Ufla, atualmente, possui apenas um caminho de transmissão, ou seja, um único canal em que os dados trafegam, como mostra a Figura 3.3, onde é evidenciado a transmissão de informações a partir do DMV.

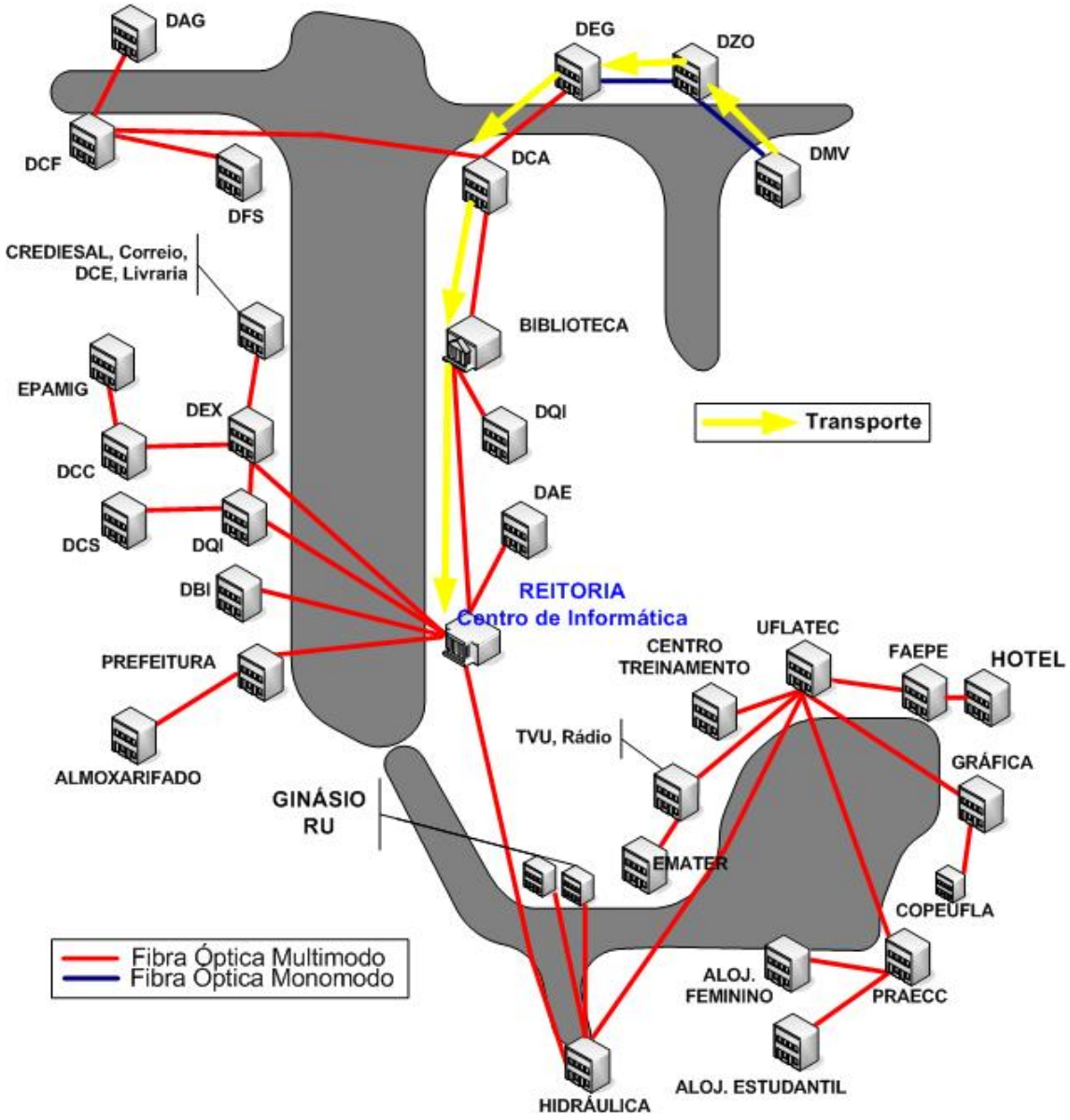


Figura 3.3: Exemplo de caminhos utilizados na transmissão de dados na Rede Ufla.

O fornecimento dos serviços de Internet parte do prédio da Reitoria, onde se localiza o Centro de Informática, para todos os outros departamentos. Analisando a Figura 3.3, podemos observar que não há caminhos redundantes para o tráfego, não apresentando, assim, tolerância à falhas. O problema que isto traz fica claro quando temos um defeito em algum equipamento (*switch*, conversor óptico, cabo ou fibra óptica). Então, caso haja uma falha, por exemplo, no *switch* do Departamento de Ciência dos Alimentos (DCA), todos aqueles departamentos que estiverem ligados ao switch do DCA – o DAG, o DCF, o DFP, o DEG, o DZO e o DMV – também ficarão com a rede inativa.

O mesmo acontece para qualquer departamento que está localizado nas extremidades do *backbone* da rede Ufla. Sempre que há problemas com equipamentos de outro departamento do qual depende a rede do primeiro, este fica com serviços de conexão indisponíveis.

O Departamento de Ciência dos Alimentos, por exemplo, é um ponto chave nesta questão. Nele estão ligados os Departamentos de Ciências Florestais, Fitopatologia, Agricultura, Engenharia, Zootecnia e Medicina Veterinária. A partir dele saem as fibras ópticas que fornecem o sinal de internet para todos estes departamentos citados. Por exemplo, os dados enviados e recebidos pelo DAG precisam passar antes pelos *switches* do DCF, DCA e Biblioteca, como mostra a Figura 3.4:

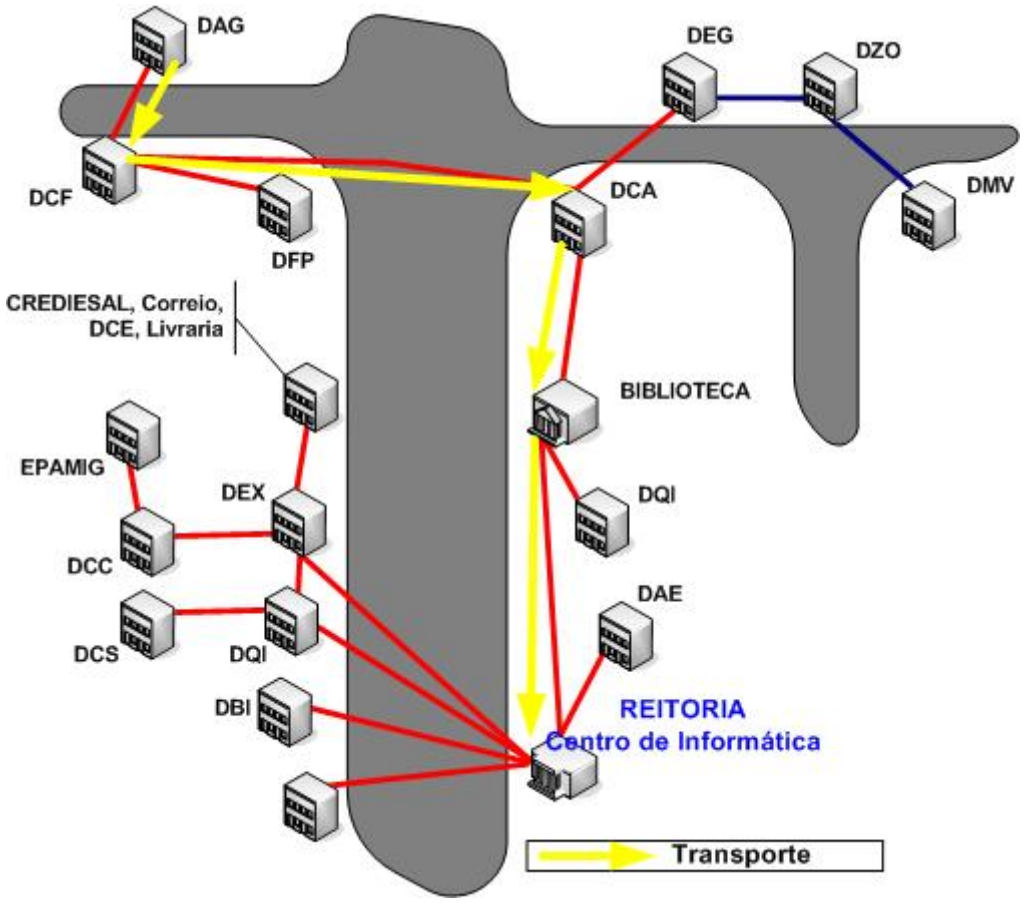


Figura 3.4: Fluxo de dados na rede entre o DAG e o *Backbone* (Cin-Ufla).

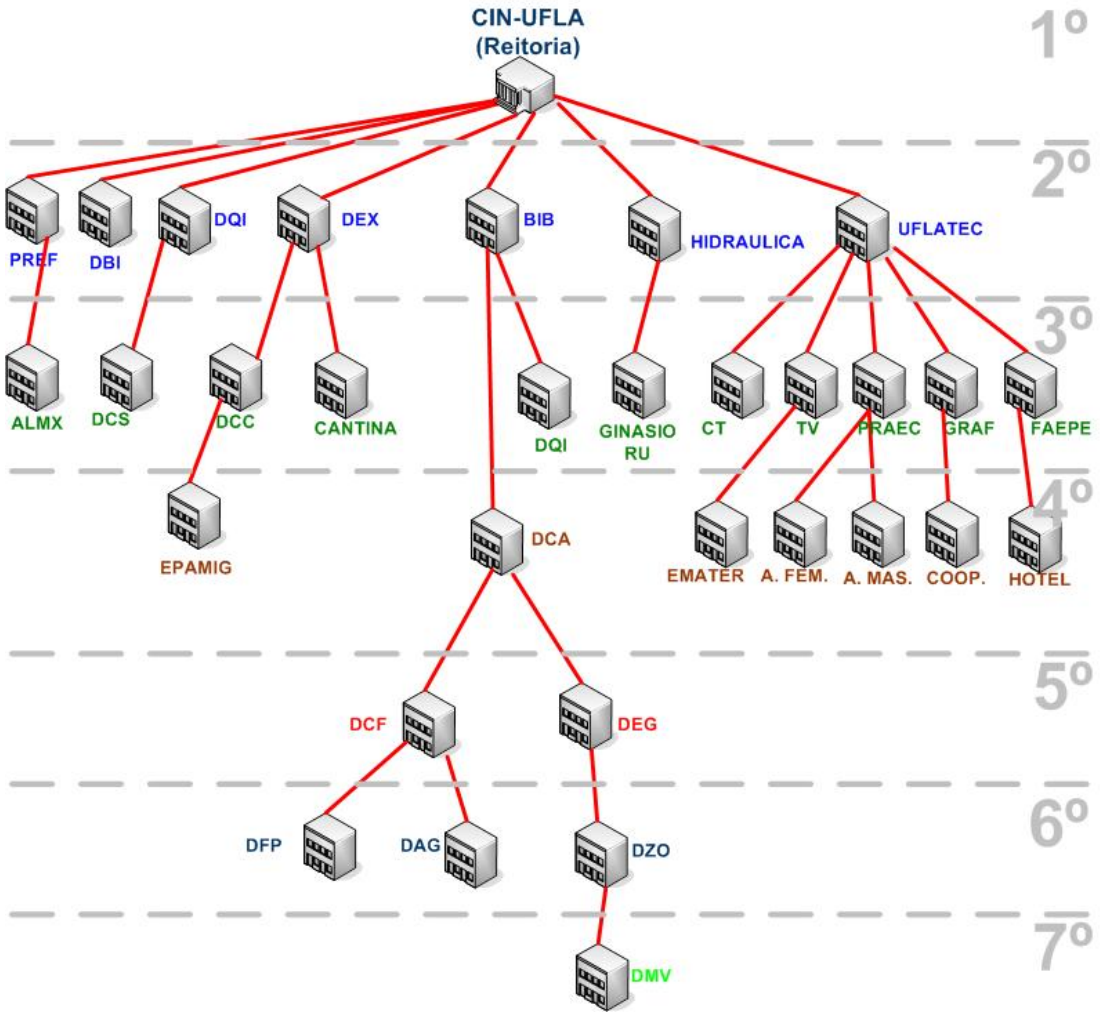


Figura 3.5: Estrutura hierárquica da Rede Ufla.

Outro local crítico para a rede Ufla é a Biblioteca, que atualmente encontra-se bastante sobrecarregada. Da Biblioteca sai somente um canal de comunicação para o DCA. Quando o enlace entre a Biblioteca e o DCA cai, todos aqueles ligados a esse também caem (DCA, DAG, DCF, DFP, DEG, DMV, DZO), como mostra a Figura 3.6:

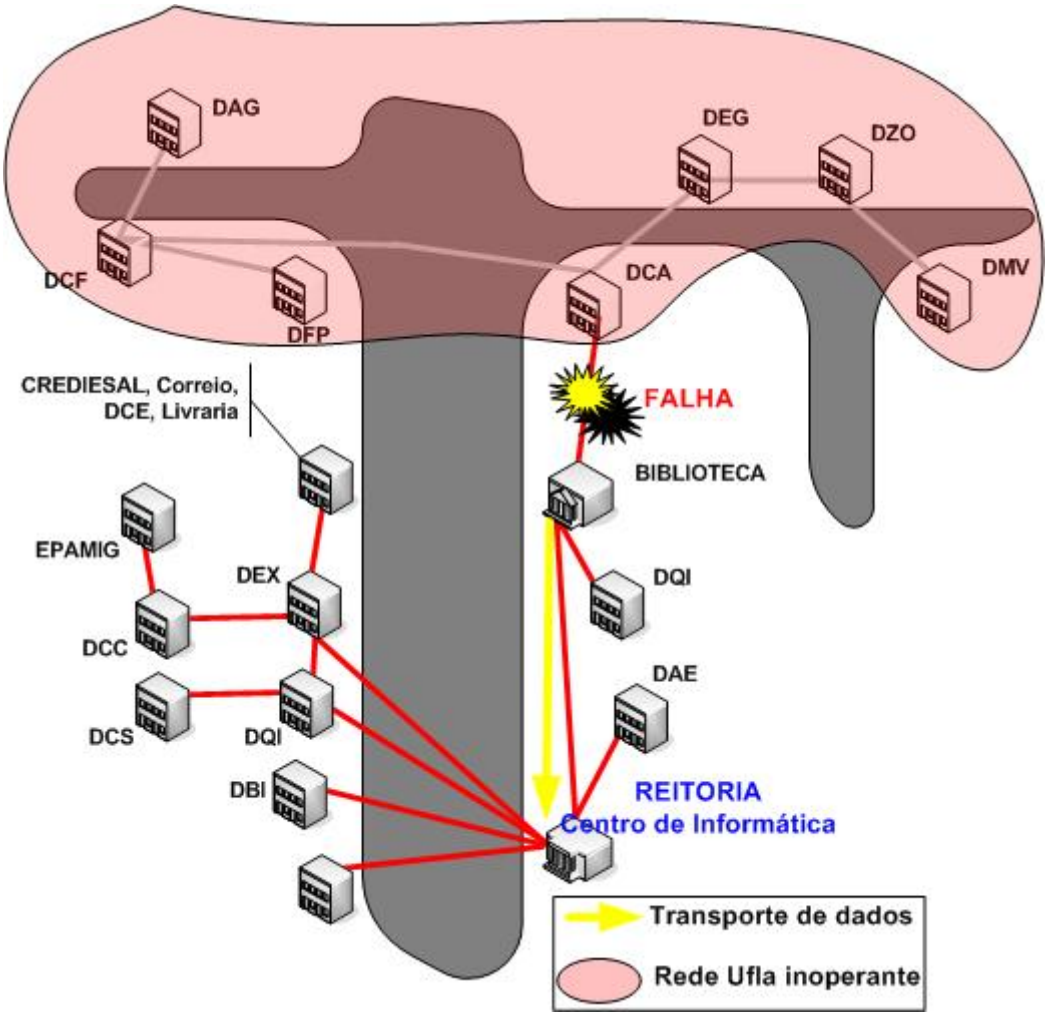


Figura 3.6: Problema da Rede em relação à Biblioteca.

Para que esses problemas possam ser solucionados, é necessário implantar redundâncias na Rede Ufla. Mais detalhes são dados na próxima seção.

3.3.2 Proposta de Redundâncias

Para que o tráfego de dados na Ufla seja eficiente, quanto mais redundâncias, melhor. Isso quer dizer que o ideal é que todos os departamentos tenham um caminho alternativo para o transporte de dados até o *backbone*. Sendo assim, diante da topologia da Rede Ufla mostrada anteriormente, foram propostas duas redundâncias:

1. Entre o Departamento de Fitopatologia (DFP) e a Cantina:
2. Entre o prédio da Reitoria e a Hidráulica.

A Figura 3.7, mostra o esquema dessas redundâncias.

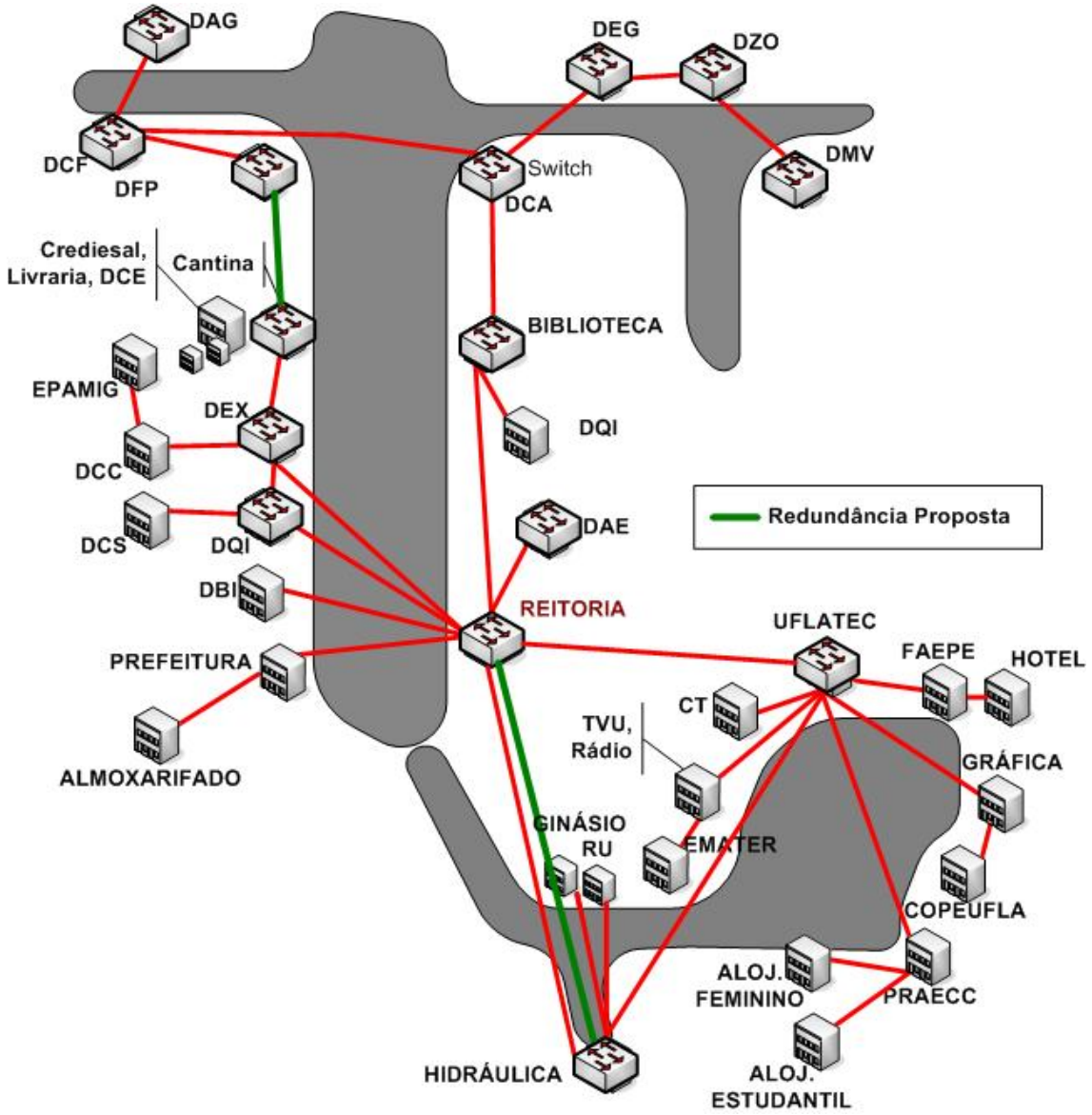


Figura 3.7: Redundâncias Propostas.

A criação de um enlace alternativo (redundância) balancearia o tráfego da rede, além de prover aos departamentos uma rota alternativa. Assim, caso haja alguma falha no link entre o DCA e a Biblioteca, por exemplo, a comunicação seria agora realizada através do novo enlace entre o DFP e a Cantina. Assim também ocorre com os prédios do Campus Histórico, caso o *link* entre os prédios da Hidráulica e o prédio da Reitoria falhe.

É importante ressaltar que a ligação entre cada departamento é feita via cabo óptico, por onde passam várias fibras (canais). Alguns departamentos apresentam canais livres podendo futuramente serem usados como novas redundâncias.

Com base nessas explicações, para realização deste plano, seriam necessários os seguintes procedimentos:

- Passagem de cabeamento óptico entre o DFP e a Cantina, entre a Reitoria e a Hidráulica e utilização de canais extras (fibras do mesmo cabo) entre o DCF e o DFP, DFP e Cantina, Reitoria e Hidráulica e Uflatec e Reitoria;
- Instalação de três novos *switches*: na Uflatec, no DFP e na Hidráulica;
- Instalação de conversores ópticos 10/100 Mbits em cada terminação das fibras, ou seja, antes de se fazer a ligação com o *switch*.
- Configuração do STP em todos os *switches* que fazem parte do anel criado (DCF, DFP, Cantina, DEX, DQI, DAE, Biblioteca, DCA, Uflatec, Hidráulica e Reitoria).

A Figura 3.8 mostra como deverá ficar a topologia da Rede Ufla após a criação dos enlaces redundantes e a instalação dos novos *switches*, com suporte à tecnologia STP.

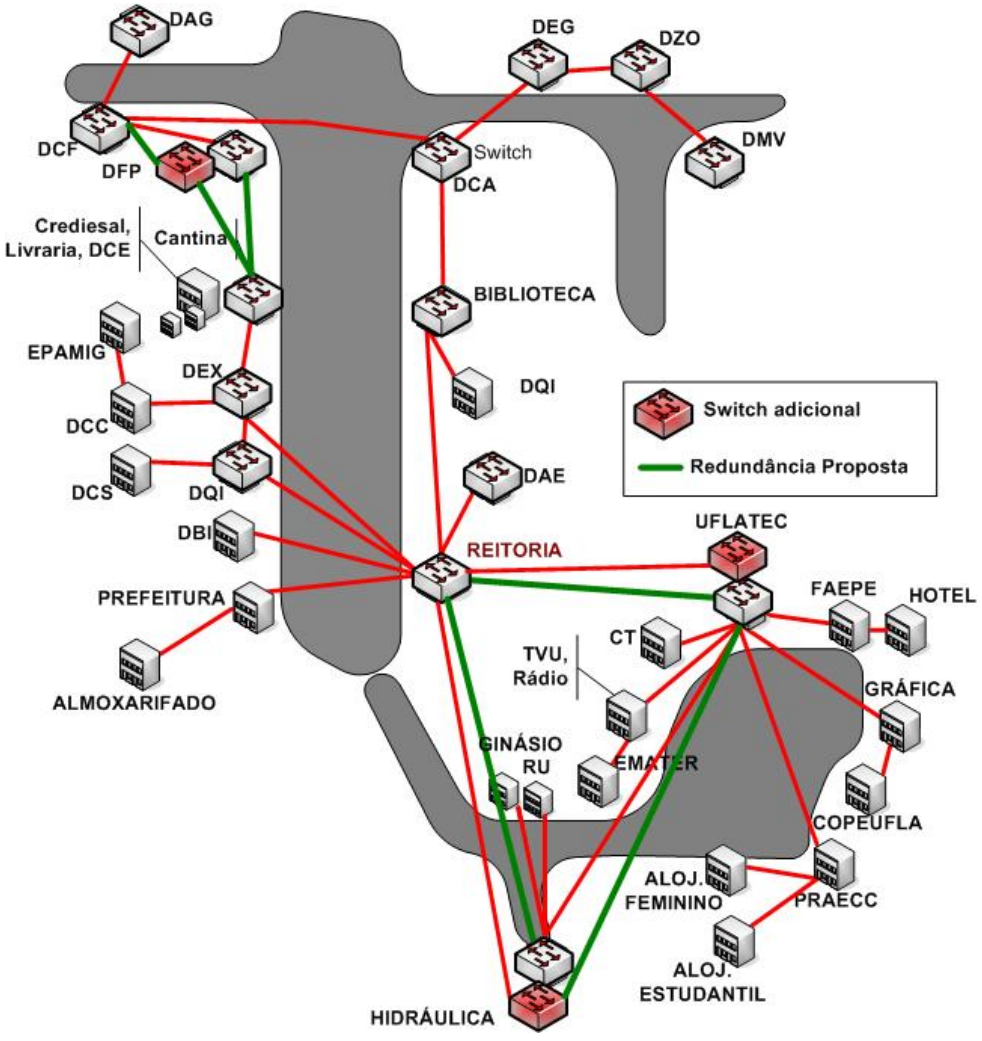


Figura 3.8: Topologia da Rede Ufla com enlaces redundantes.

Problemas gerados pela implantação de redundância

A presença de redundância é extremamente importante nas redes, principalmente em uma rede do tipo Campus, como é o caso da Rede Ufla, onde diversas informações importantes são transmitidas e a disponibilidade da rede deve ser a maior possível.

Enlaces alternativos oferecem uma maior disponibilidade e tolerância à falhas de uma rede. Quando algum enlace falha, a existência de um caminho alternativo garante que a comunicação não vai ser interrompida por um longo tempo.

Apesar desta vantagem, o uso de redundâncias físicas interligadas por switches gera grandes problemas. Os principais são os loops e as tempestades de *broadcast* (*Broadcast Storm*), que prejudicam consideravelmente o desempenho de uma rede. A Figura 3.9 mostra uma rede contendo loops.

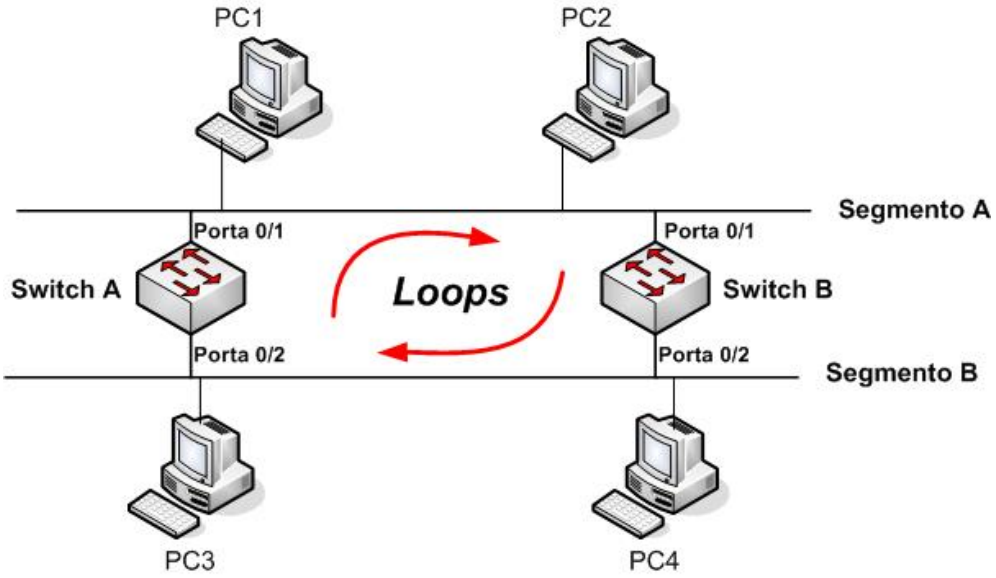


Figura 3.9: *Loops* em uma rede redundante.

Para exemplificar o que acontece com os dados em uma rede redundante (sem protocolos adicionais), suponhamos que o PC 1 queira enviar dados para o PC 3, e que ambos Switches A e B ainda não possuem a informação sobre o endereço MAC do PC 3 em suas respectivas tabelas de endereços MAC. Uma vez que a localização física (porta do switch) onde PC 3 está conectado é desconhecida por ambos *switches* A e B, o pacote será enviado para todas as portas (*broadcast*), exceto à porta de origem.

Ambos *switches* "aprendem" o endereço MAC do PC 1, fazendo a devida referência em suas respectivas tabelas. Os dois *switches* enviam o quadro para o segmento B, pois

ainda não possuem a informação sobre a localização física do PC 3 (ou seja, estes *switches* desconhecem o endereço MAC do PC 3).

Assim o problema ocorre: uma vez que ambos *switches* enviam o quadro para o segmento B, o switch A receberá o quadro enviado pelo Switch B, e o Switch B receberá o quadro enviado pelo Switch A.

No início de todo este processo, ambos *switches* A e B tinham como referência as suas portas 0/1 para o endereço MAC do PC 1, mas, neste exato momento, eles acabam de receber um quadro indicando que o endereço MAC de origem (do PC 1) está localizado nas suas respectivas portas 0/2. Os *switches* sabiam que PC 1 estava conectado na porta 0/1, e esta referência estava associada em suas respectivas tabelas (o endereço MAC do PC 1, porta de origem 0/1).

Uma vez que o endereço MAC de origem do quadro Ethernet indica que o PC 1 (seu endereço MAC) está passando pela porta 0/2, os *switches* serão obrigados a refazer o conteúdo de suas tabelas de endereçamento, referente ao endereço MAC do PC 1. Ao invés de associar este endereço MAC à porta 0/1, os switches o farão para as suas respectivas portas 0/2.

O quadro é devolvido para o segmento A, pois o PC 4 ainda não o recebeu. Ao chegar ao segmento A, o quadro será recebido por ambos switches, nas mesmas condições citadas anteriormente. Switch A receberá o quadro enviado pelo Switch B, e o B de A. O próximo passo é enviar o quadro novamente para o segmento B. Assim, este processo ocorrerá indefinidamente, sendo denominado de "*Bridging loop*".

A situação ficaria muito mais caótica caso o pacote enviado pelo PC 1 fosse um quadro de *broadcast*, que é destinado a todas as máquinas. Isto elevaria a condição do problema para o que chamamos de Tempestade de *Broadcast* ou *Broadcast Storm*, o que seria muito pior em termos de performance e utilização dos recursos da rede.

Segundo KUROSE & ROSS (2003), tudo isso pode ser evitado fazendo uso da técnica de *Spanning Tree Protocol*. Os procedimentos realizados para viabilizar a implantação desse mecanismo estão explicados a seguir.

3.3.3 Proposta de Implantação do STP

Conforme citado na seção 2.8, o STP é responsável pela eliminação dos *loops* e Tempestades de *Broadcast*, gerenciando os caminhos redundantes e oferecendo uma alta disponibilidade para a rede.

Para a implantação do STP na rede Ufla, utilizando as redundâncias sugeridas na seção anterior, todos os switches que fazem parte da topologia redundante (anel formado pela interligação dos *switches*) teriam que ser reconfigurados.

A implantação do STP será mostrada nesta seção tomando como exemplo a redundância entre o Departamento de Fitopatologia e a Cantina. Deverão ser feitos três procedimentos iniciais: 1) instalar um novo *switch* no DFP, 2) usar um canal extra do cabo óptico já existente entre o DCF e o DFP para fazer uma redundância entre o DFP e o DCF, 3) passar de um novo cabo de fibra entre o DFP e a Cantina, usando dois canais dessa fibra para fazer a redundância entre o DFP e a Cantina e 4) Conectar um canal ao Switch1 e outro canal ao Switch2.

Os dois *switches* do DFP, portanto, serão ligados à Cantina através de dois canais da nova fibra óptica que será lançada (canal redundante). Na Figura 3.10, como um dos *switches* do DFP situa-se em um local apropriado (centro), ele terá uma prioridade menor em relação aos outros, o que garante sua vitória na eleição de *Switch Root* (Raiz) da topologia STP.

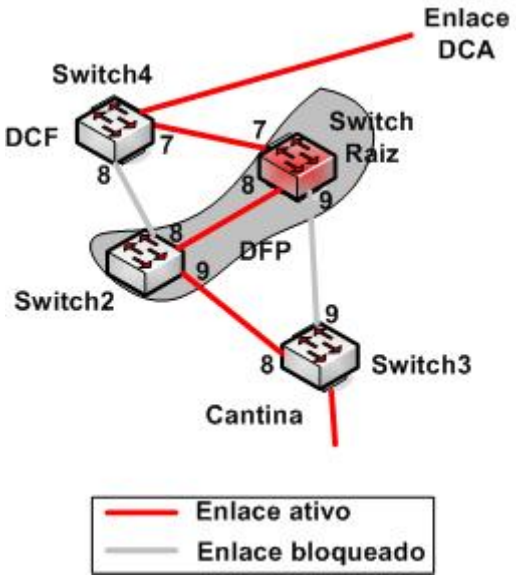


Figura 3.10: Implantação do STP no anel entre o DCF e a Cantina.

As portas 8 do Switch4 e 9 do Switch3, previamente configuradas com uma prioridade secundária propositalmente, ficarão em estado de *backup* (bloqueadas), até que algum problema aconteça e estas passem a ser primárias.

Para escolher o caminho que inicialmente será o primário (ativo) basta dar valores menores às prioridades das portas dos *switches*. Isso foi feito para a porta 7 do Switch4, portas 8 e 9 do Switch2 e para a porta 8 do Switch3. O Switch Raiz tem todas suas portas ativas, por determinação do STP.

Na Figura 3.11, são mostradas as redundâncias propostas e também como ficaria o tráfego da rede após a implantação do STP.

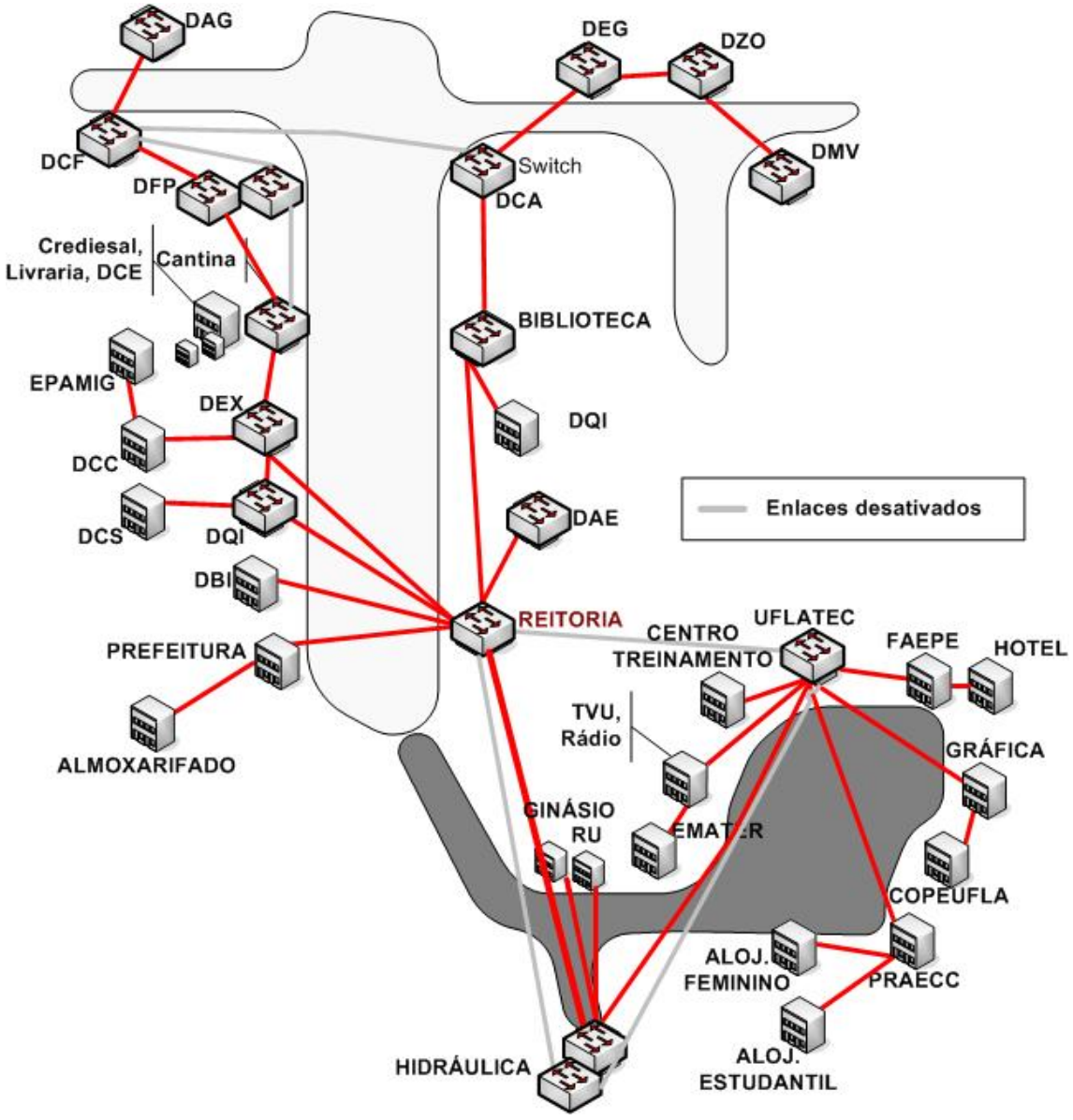


Figura 3.11: Topologia ativa e enlaces desabilitados pelo STP.

Teste Piloto

Para a verificação e validação da técnica proposta, foi realizado um teste piloto, que consistiu em montar um laboratório para simular o funcionamento do *Spanning Tree Protocol* em alguns *switches*.

Utilizou-se dois *switches* Ethernet Planet WGSW1602, cedidos pelo Centro de Informática da Ufla (Cin-Ufla).

Existem dois tipos de *switches*: os gerenciáveis e os não-gerenciáveis. Nos não-gerenciáveis, a configuração é automática, ou seja, não há interface entre administrador e equipamento. Já nos *switches* gerenciáveis, existe um *software* que permite a manipulação de funções, como VLAN, STP, monitoramento de tráfego, etc. Os equipamentos utilizados eram gerenciáveis, por isso foi possível a configuração do *Spanning Tree*.

O *software* de gerenciamento é um utilitário que já vem incluso no firmware do switch e pode ser acessado via http através de um navegador (*browser*), como mostra a Figura 3.12.

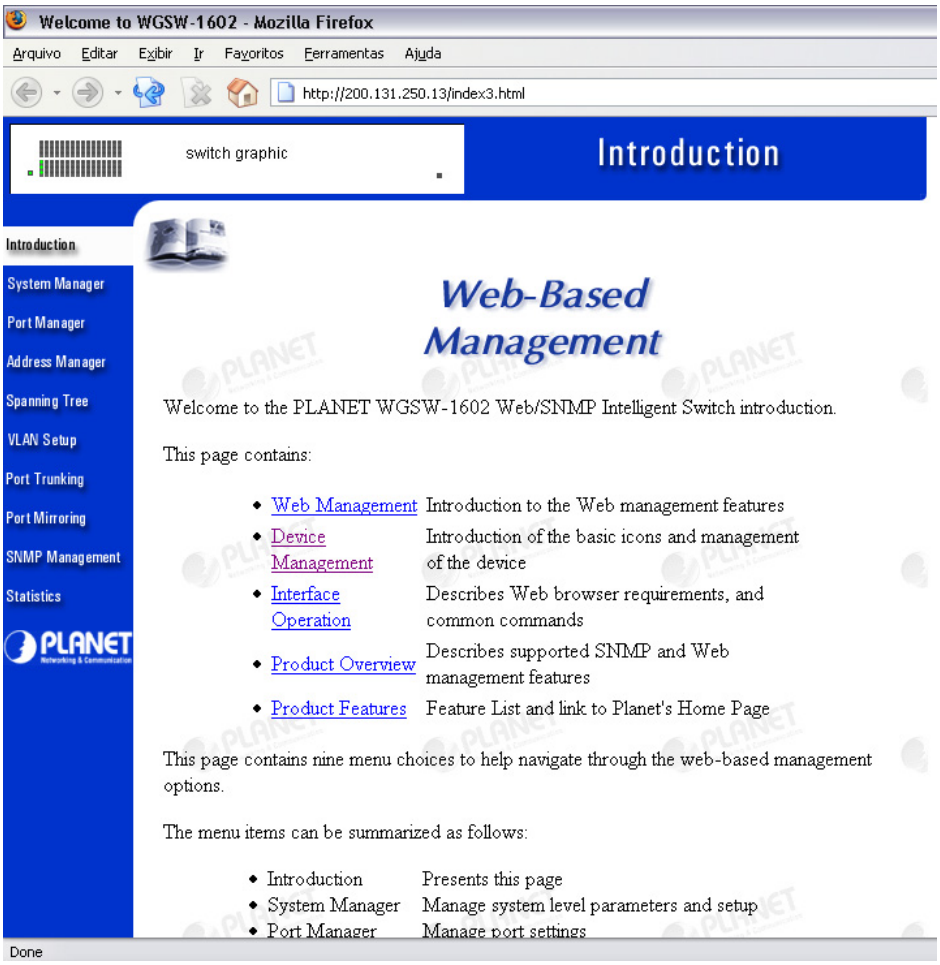


Figura 3.12: Tela inicial de configuração do Switch Planet WGSW 1602.

Os *switches* foram ligados formando uma topologia redundante, como mostra a Figura 3.13. Devido ao fato de haver somente dois equipamentos disponíveis, a topologia montada em laboratório pode ser entendida como uma redundância mínima.

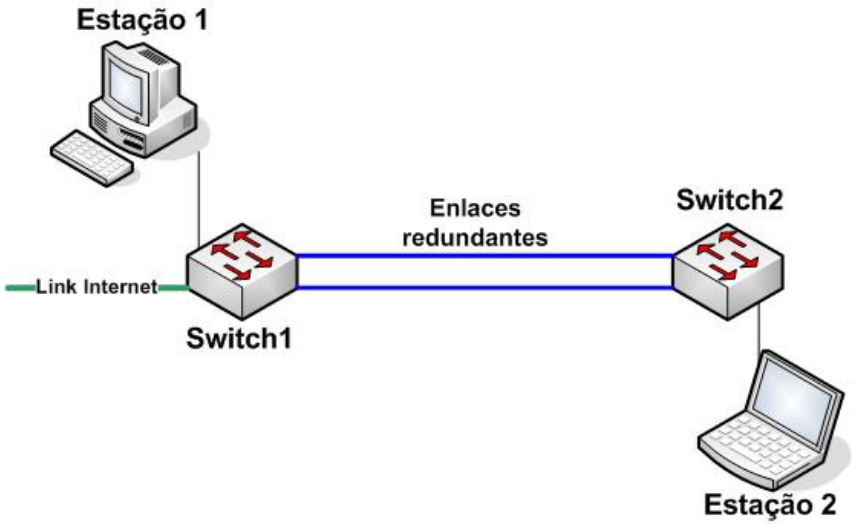


Figura 3.13: *Switches* com enlaces redundantes.

Na ativação do STP, ambos os *switches* tiveram que ser configurados. Esta configuração foi feita via navegador (*browser*), pelo utilitário de configuração do *switch*, que oferece a possibilidade de alterar quaisquer funções que o acompanham. A configuração consistiu em ativar o STP e fazer ajustes nos "*timers*" ou tempos de execução do protocolo.

O acesso a cada *switch* é feito utilizando seu número IP. Para que eles pudessem se comunicar, foi necessário alterar esses IPs de modo que fizessem parte da mesma rede. Essa alteração foi feita dentro da opção *System Manager*, usando a opção *IP Settings*. Após as modificações, os dois switches apresentaram as seguintes características:

Tabela 3.2: Configuração dos switches.

	Switch1	Switch2
IP	200.131.250.13	200.131.250.252
Máscara	255.255.255.0	255.255.255.0
MAC	00:30:4F:00:04:0E	00:30:4F:00:04:0E

Ao computador foi atribuído um IP pertencente à mesma classe dos IPs dos switches, conforme a Tabela 3.3.

Tabela 3.3: Configuração do computador.

	PC 1
IP	200.131.250.1
Máscara	255.255.255.0

O STP foi configurado de acordo com o padrão IEEE, que determina valores específicos para os *timers*:

Tabela 3.4: Tempos utilizados na configuração do STP.

Hello Time	2 segundos
Max Age	20 segundos
Forward Delay	15 segundos
Bridge Priority	32768

A cada porta foi associado o custo 19 previsto pelo IEEE para redes 100 Mbps. Os enlaces redundantes foram ligados utilizando as portas 7 e 8 de cada *switch*. Ligou-se a porta 7 do Switch1 à porta 7 do Switch2, e a porta 8 do Switch1 à porta 8 do Switch2.

Convencionou-se que a porta 7 seria o enlace primário, tendo assim prioridade 1, em relação à porta 8, que teve prioridade 128. Isto garantiu que o *link* usado inicialmente, até que houvesse alguma falha, fosse o que trafegava dados através da porta 7, como mostra a Figura 3.14:

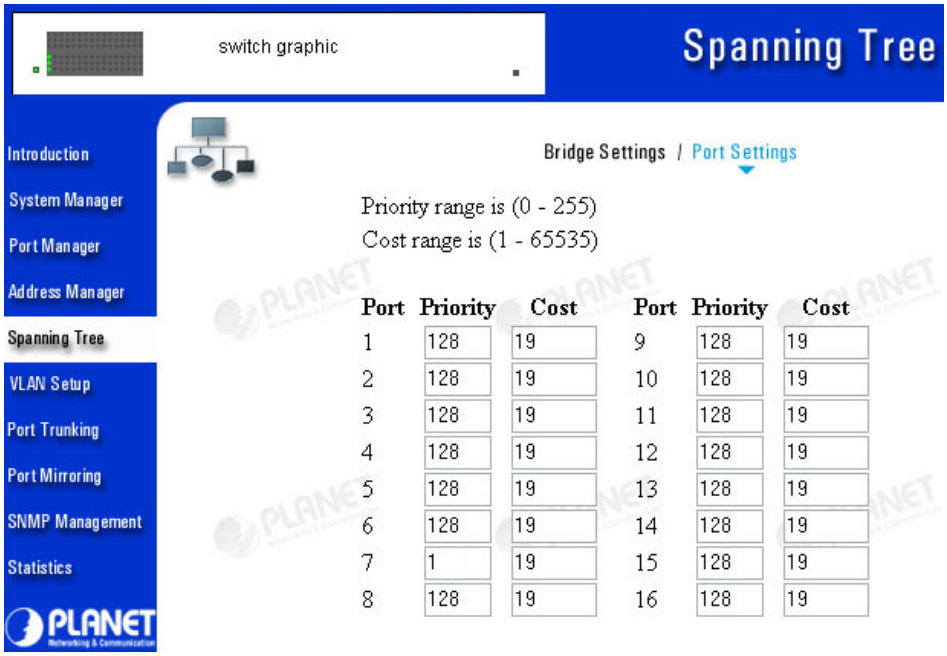


Figura 3.14: Custos e prioridades de cada porta.

Para avaliação e monitoramento da execução do *Spanning Tree*, um *notebook* foi conectado ao Switch 2 (200.131.250.252) e ao Switch 1 (200.131.250.13) um *link* de internet proveniente de um servidor. A estação 2 foi usada para monitorar o funcionamento do STP e o tráfego da rede.

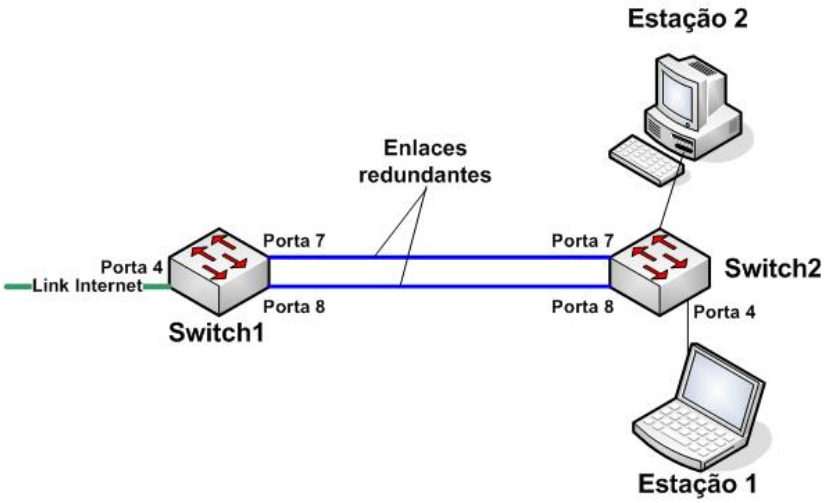


Figura 3.15: Ambiente Simulado.

Foi utilizado o Ping *(Packet Internet Groper*), um programa usado para testar o alcance de uma rede, que envia a nós remotos uma requisição e espera por uma resposta. Através dele, foi possível também identificar o momento em que o enlace ficou indisponível e quando voltou à atividade.

Na topologia construída, foi realizada a simulação da queima de uma porta do *switch*, que, em uma rede normal, acarretaria na interrupção do *link*. Uma analogia desta operação pode ser feita com a queima de conversores ópticos.

Os resultados dos testes e de todo o trabalho são mostrados no Capítulo seguinte (Resultados e Discussões).

4 RESULTADOS E DISCUSSÕES

O primeiro resultado deste trabalho foi verificar que a Rede Ufla apresenta, há bastante tempo, um problema de tráfego de dados que prejudica consideravelmente o desempenho dos trabalhos, realizados na Universidade, que precisam utilizar a rede constantemente. O fluxo de dados e informações mostra-se bastante intenso e a rede não possui trajetos redundantes para aumentar a confiabilidade da mesma.

Como é crescente o número de computadores utilizados na Ufla, o tráfego da rede torna-se bastante intenso e as falhas são inevitáveis. Entre as causas dessas falhas, estão o travamento de um *switch* e a queima de um conversor óptico. Um dos motivos que levam os s*witches* a travar é quando uma grande quantidade de dados trafega através de suas portas, provocando a sobrecarga. Um conversor óptico pode queimar quando houver quedas na rede elétrica.

Após a montagem do laboratório e feitas as devidas configurações dos *switches*, realizou-se vários testes, a fim de simular a topologia real da Rede Ufla e o funcionamento do *Spanning Tree* em enlaces redundantes. Os testes realizados e seus propósitos são explicados a seguir:

Comportamento de uma rede com redundância sem o STP

Um importante teste realizado foi o do comportamento de uma rede redundante sem a implantação do STP. Nesta simulação, a mesma topologia com enlaces alternativos foi utilizada, porém, sem a ativação do STP.

No monitoramento de portas realizado através do próprio software de gerenciamento do *switch*, a opção *Port Manager* mostrou que todas as portas de ambos os switches estavam encaminhando quadros. Tanto a porta definida como “Principal” quanto a “*Backup*” encaminhavam dados (Fwd), como mostra a Figura 4.1. Isso significa que as informações trafegavam livremente através dos dois enlaces redundantes, sobrecarregando a rede e gerando *loops*.

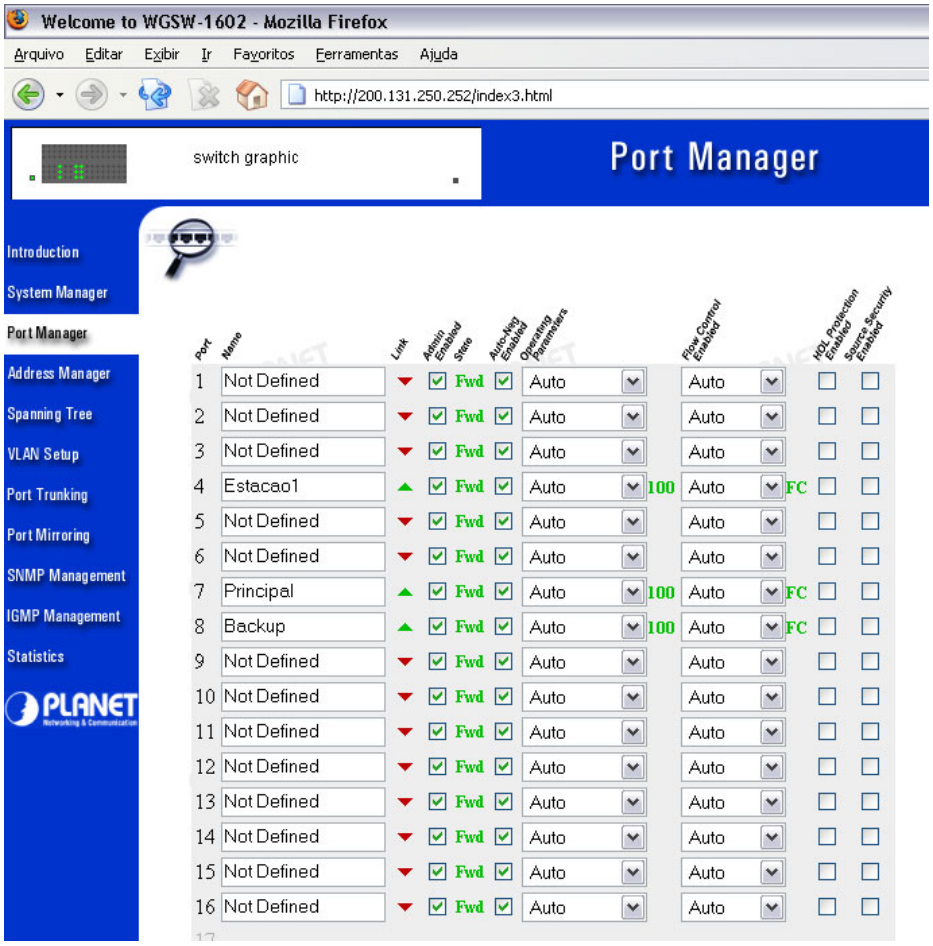
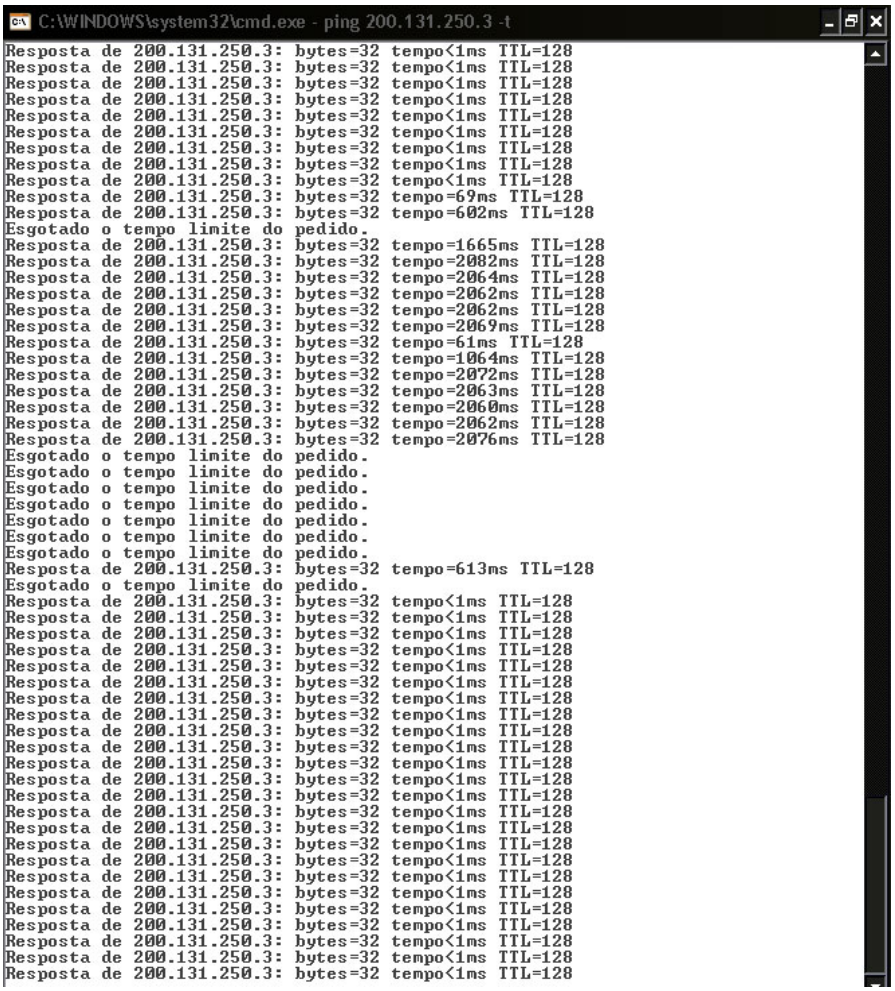


Figura 4.1: Estado das portas do switch.

Observou-se que à medida em que o tráfego da rede tornava-se mais intenso, os pacotes demoravam um tempo maior para chegarem ao destino. A Figura 4.2 ilustra a linha do tempo da transmissão dos pacotes:



```
Resposta de 200.131.250.3: bytes=32 tempo<1ms TTL=128
Resposta de 200.131.250.3: bytes=32 tempo<1ms TTL=128
Resposta de 200.131.250.3: bytes=32 tempo<1ms TTL=128
Resposta de 200.131.250.3: bytes=32 tempo<1ms TTL=128
Resposta de 200.131.250.3: bytes=32 tempo<1ms TTL=128
Resposta de 200.131.250.3: bytes=32 tempo<1ms TTL=128
Resposta de 200.131.250.3: bytes=32 tempo<1ms TTL=128
Resposta de 200.131.250.3: bytes=32 tempo<1ms TTL=128
Resposta de 200.131.250.3: bytes=32 tempo<1ms TTL=128
Resposta de 200.131.250.3: bytes=32 tempo=69ms TTL=128
Resposta de 200.131.250.3: bytes=32 tempo=602ms TTL=128
Esgotado o tempo limite do pedido.
Resposta de 200.131.250.3: bytes=32 tempo=1665ms TTL=128
Resposta de 200.131.250.3: bytes=32 tempo=2082ms TTL=128
Resposta de 200.131.250.3: bytes=32 tempo=2064ms TTL=128
Resposta de 200.131.250.3: bytes=32 tempo=2062ms TTL=128
Resposta de 200.131.250.3: bytes=32 tempo=2062ms TTL=128
Resposta de 200.131.250.3: bytes=32 tempo=2069ms TTL=128
Resposta de 200.131.250.3: bytes=32 tempo=61ms TTL=128
Resposta de 200.131.250.3: bytes=32 tempo=1064ms TTL=128
Resposta de 200.131.250.3: bytes=32 tempo=2072ms TTL=128
Resposta de 200.131.250.3: bytes=32 tempo=2063ms TTL=128
Resposta de 200.131.250.3: bytes=32 tempo=2060ms TTL=128
Resposta de 200.131.250.3: bytes=32 tempo=2062ms TTL=128
Resposta de 200.131.250.3: bytes=32 tempo=2076ms TTL=128
Esgotado o tempo limite do pedido.
Esgotado o tempo limite do pedido.
Esgotado o tempo limite do pedido.
Esgotado o tempo limite do pedido.
Esgotado o tempo limite do pedido.
Esgotado o tempo limite do pedido.
Esgotado o tempo limite do pedido.
Resposta de 200.131.250.3: bytes=32 tempo=613ms TTL=128
Esgotado o tempo limite do pedido.
Resposta de 200.131.250.3: bytes=32 tempo<1ms TTL=128
Resposta de 200.131.250.3: bytes=32 tempo<1ms TTL=128
Resposta de 200.131.250.3: bytes=32 tempo<1ms TTL=128
Resposta de 200.131.250.3: bytes=32 tempo<1ms TTL=128
Resposta de 200.131.250.3: bytes=32 tempo<1ms TTL=128
Resposta de 200.131.250.3: bytes=32 tempo<1ms TTL=128
Resposta de 200.131.250.3: bytes=32 tempo<1ms TTL=128
Resposta de 200.131.250.3: bytes=32 tempo<1ms TTL=128
Resposta de 200.131.250.3: bytes=32 tempo<1ms TTL=128
Resposta de 200.131.250.3: bytes=32 tempo<1ms TTL=128
Resposta de 200.131.250.3: bytes=32 tempo<1ms TTL=128
Resposta de 200.131.250.3: bytes=32 tempo<1ms TTL=128
Resposta de 200.131.250.3: bytes=32 tempo<1ms TTL=128
Resposta de 200.131.250.3: bytes=32 tempo<1ms TTL=128
Resposta de 200.131.250.3: bytes=32 tempo<1ms TTL=128
Resposta de 200.131.250.3: bytes=32 tempo<1ms TTL=128
Resposta de 200.131.250.3: bytes=32 tempo<1ms TTL=128
Resposta de 200.131.250.3: bytes=32 tempo<1ms TTL=128
Resposta de 200.131.250.3: bytes=32 tempo<1ms TTL=128
Resposta de 200.131.250.3: bytes=32 tempo<1ms TTL=128
Resposta de 200.131.250.3: bytes=32 tempo<1ms TTL=128
Resposta de 200.131.250.3: bytes=32 tempo<1ms TTL=128
Resposta de 200.131.250.3: bytes=32 tempo<1ms TTL=128
Resposta de 200.131.250.3: bytes=32 tempo<1ms TTL=128
```

Figura 4.2: Status da comunicação sem STP.

Como mostra a Figura 4.2, a comunicação entre os computadores em alguns momentos tornou-se muito lenta (alto valor do tempo de resposta), chegando até a não mais conseguir uma resposta em tempo hábil. Tais acontecimentos evidenciam a ocorrência de *loops* entre os dois *switches*, fazendo com que suas tabelas de endereçamento ficassem confusas.

Dentre a série de testes realizados sem o STP, notou-se que quando o enlace que estava sendo usado para trafegar dados foi bloqueado propositalmente, a comunicação não foi restabelecida, provando assim que a não configuração do STP em uma rede redundante pode fazer com que a mesma não consiga se restabelecer após a ocorrência de uma falha.

A Figura 4.3 mostra uma estatística dos dados trafegados em cada porta, enquanto o *Spanning Tree* não estava presente:

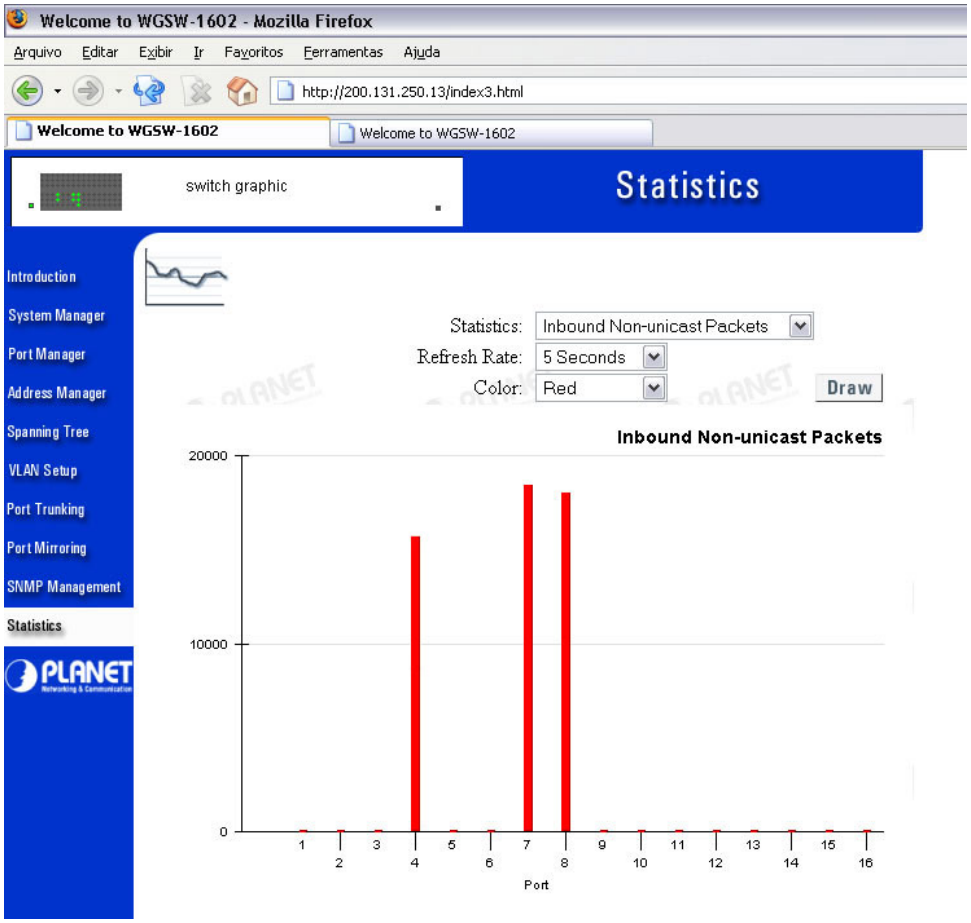


Figura 4.3: Fluxo de *Broadcast* sem STP.

Os pacotes *broadcast* enviados pelo *host* de origem chegavam a todas as portas onde havia redundância. Ou seja, o pacote trafegava tanto pelo enlace Principal (Porta 7) quanto pelo link de *Backup* (Porta 8). Com isso, os *switches* tinham comprometimento em suas tabelas de endereçamento e o exagerado fluxo de dados prejudicava o desempenho das transferências.

Um acontecimento relevante também observado foi a instabilidade da rede. Em alguns momentos o computador (Estação 1) não conseguia acessar a rede interna, apenas a externa. Este fato alternou-se durante vários momentos.

A razão disso é a confusão das tabelas de endereçamento dos switches, gerada pelo excesso de quadros trafegados. Como dois caminhos estavam sendo usados para comunicação, alguns quadros precisavam ser retransmitidos, aumentando o tempo de resposta de cada requisição.

Observando este problema, podemos ter uma idéia do que acontece se o STP não for configurado na rede Ufla após implantação das redundâncias. Como é uma rede de porte muito grande, seu desempenho seria extremamente prejudicado.

Comportamento de uma rede com redundância utilizando o STP

Após ativação e configuração do *Spanning Tree* em cada *switch*, procedeu a etapa de testes e monitoramento.

As estatísticas das portas revelaram o efetivo funcionamento do STP, bloqueando a redundância e utilizando somente a porta escolhida como “Principal” para enviar e receber os dados. A Figura 4.4 mostra o status das portas. Podemos observar que a porta 8 (Enlace de *Backup*) estava bloqueada (Blk), impedindo que qualquer tipo de pacote fosse encaminhado através dela.

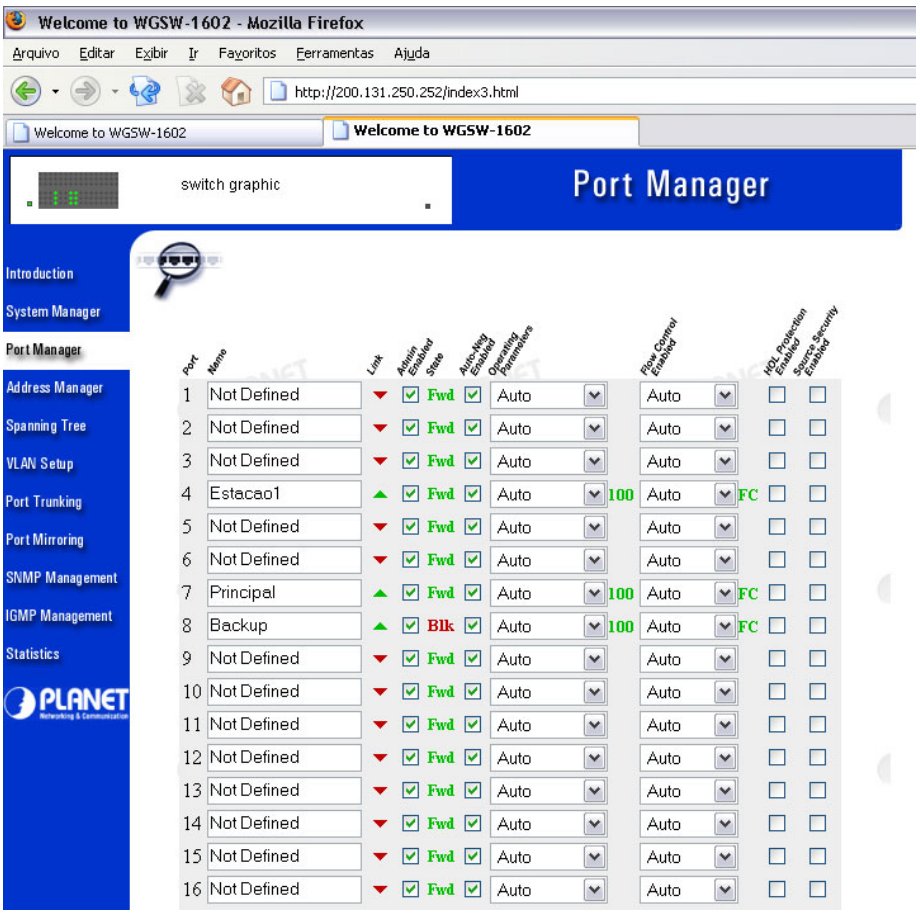


Figura 4.4: Status das portas com STP.

A avaliação do funcionamento do STP foi feita ao desconectar um dos cabos que ligava os switches, simulando a queda de um link, e verificar se a comunicação foi restabelecida e mantida.

As Figuras 4.5 e 4.6 mostram o que aconteceu após a retirada de um cabo que interligava os switches:

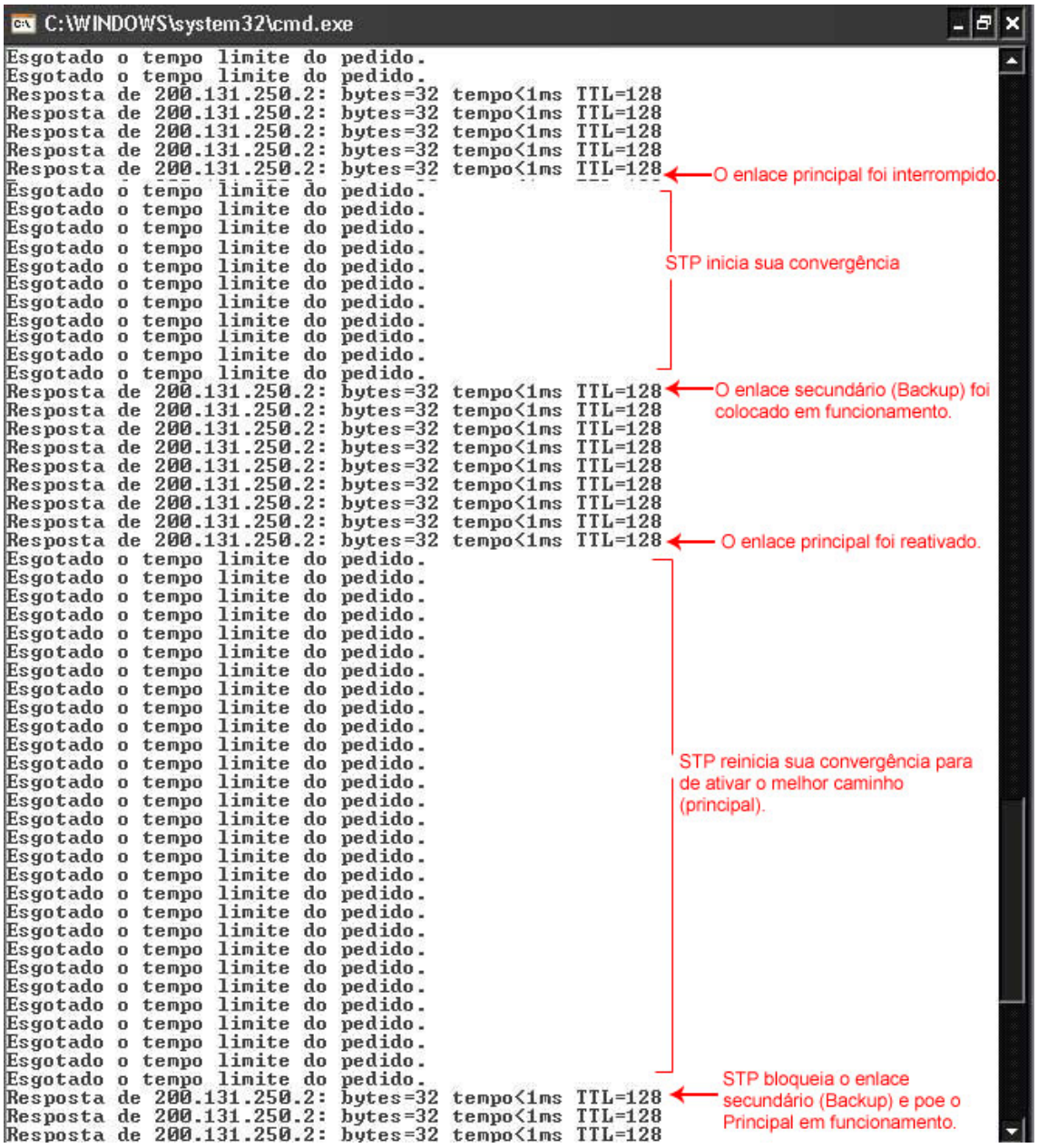

C:\WINDOWS\system32\cmd.exe
Esgotado o tempo limite do pedido.
Esgotado o tempo limite do pedido.
Resposta de 200.131.250.2: bytes=32 tempo<1ms TTL=128
Resposta de 200.131.250.2: bytes=32 tempo<1ms TTL=128
Resposta de 200.131.250.2: bytes=32 tempo<1ms TTL=128
Resposta de 200.131.250.2: bytes=32 tempo<1ms TTL=128
Resposta de 200.131.250.2: bytes=32 tempo<1ms TTL=128
Esgotado o tempo limite do pedido.
Esgotado o tempo limite do pedido.
Esgotado o tempo limite do pedido.
Esgotado o tempo limite do pedido.
Esgotado o tempo limite do pedido.
Esgotado o tempo limite do pedido.
Esgotado o tempo limite do pedido.
Esgotado o tempo limite do pedido.
Esgotado o tempo limite do pedido.
Esgotado o tempo limite do pedido.
Esgotado o tempo limite do pedido.
Resposta de 200.131.250.2: bytes=32 tempo<1ms TTL=128
Resposta de 200.131.250.2: bytes=32 tempo<1ms TTL=128
Resposta de 200.131.250.2: bytes=32 tempo<1ms TTL=128
Resposta de 200.131.250.2: bytes=32 tempo<1ms TTL=128
Resposta de 200.131.250.2: bytes=32 tempo<1ms TTL=128
Resposta de 200.131.250.2: bytes=32 tempo<1ms TTL=128
Resposta de 200.131.250.2: bytes=32 tempo<1ms TTL=128
Resposta de 200.131.250.2: bytes=32 tempo<1ms TTL=128
Esgotado o tempo limite do pedido.
Esgotado o tempo limite do pedido.
Esgotado o tempo limite do pedido.
Esgotado o tempo limite do pedido.
Esgotado o tempo limite do pedido.
Esgotado o tempo limite do pedido.
Esgotado o tempo limite do pedido.
Esgotado o tempo limite do pedido.
Esgotado o tempo limite do pedido.
Esgotado o tempo limite do pedido.
Esgotado o tempo limite do pedido.

Figura 4.6: Fluxo de pacotes em cada porta.

Na Figura 4.6, a comunicação ainda estava usando a porta 7. Pôde-se observar que não havia informações trafegando pela porta 8, que permanecia bloqueada. O STP garantiu a entrega dos dados sem provocar problemas de performance, uma vez que não deixava os pacotes serem transmitidos pelas duas portas.

Após a queda do enlace “Principal”, o STP levou em média um minuto e meio para convergir. Isto é, perceber que havia a queda de um canal de comunicação e remontar a topologia da árvore de cobertura, colocando a porta que estava bloqueada (Backup) no estado de encaminhamento, suprindo assim a falha do outro enlace.

A Figura 4.7 mostra o fluxo de dados, agora utilizando a porta 8.

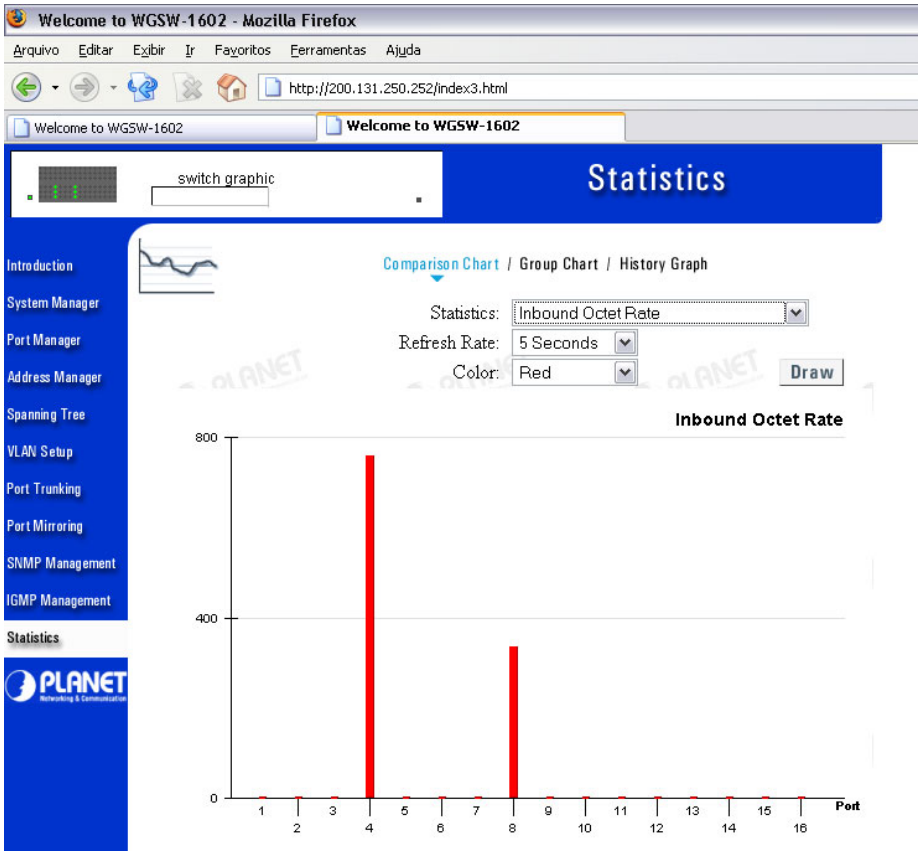


Figura 4.7: Comunicação utilizando a porta de Backup.

Uma outra importante característica do STP observada foi a capacidade de aprendizagem devido ao número de falhas ocorridas. Na primeira vez em que o enlace foi interrompido, o STP levou mais tempo para convergir em comparação às ocorrências de falhas posteriores.

Para a implantação dessa técnica na Rede Ufla, além das redundâncias, deverão ser instalados *switches* adicionais em cada local onde fossem lançadas novas fibras. Com isso, cada canal de fibra estaria ligado a um *switch*, provendo assim uma segurança ainda maior.

Se switches adicionais não fossem usados, problemas como a queima desses equipamentos, que são bastante comuns, não teriam solução, visto que o único *switch* que estaria sendo usado na comunicação estaria inoperante.

Quando múltiplos *switches* são usados, além da possibilidade de convergência, em caso da falha física de enlace (como a quebra de uma fibra), o sistema suportaria também a falha deste equipamento.

Diante do exposto, o presente trabalho elaborou uma proposta de otimização do tráfego de dados na Rede Ufla, através da implantação da técnica de *Spanning Tree*

Protocol nos equipamentos responsáveis pela convergência dos links departamentais – os *switches*.

O STP se mostrou bastante eficiente quanto à sua convergência, visto que gastou em média de um a dois minutos para restabelecer a conexão entre as estações. É um tempo considerável, se comparado ao tempo que a rede Ufla fica indisponível por motivos de falha de um *switch* ou outro equipamento.

Para otimizar ainda mais o tempo de convergência do STP, pode se fazer uso de *switches* com suporte à tecnologia *Rapid Spanning Tree Protocol* (RSTP). Como visto na seção 2.9, o RSTP diminui significativamente o tempo de convergência após uma falha. Porém, essa nova versão do STP não poderá ser implantada na Ufla, pois todos os *switches* utilizados em sua rede não têm suporte ao RSTP.

5 CONCLUSÃO

O projeto surgiu da necessidade de se otimizar a topologia e principalmente o desempenho da rede da Universidade Federal de Lavras. Soluções previamente testadas várias vezes e de diversas maneiras em laboratório foram propostas a fim de prover um aumento na disponibilidade e performance da rede.

A prévia análise da rede Ufla demonstrou que a mesma não apresenta redundâncias ao longo de sua topologia. E como é uma rede constantemente usada e de tráfego bastante intenso, muitos problemas podem ocorrer, como é o caso da falha de um switch.

De acordo com esta análise, e considerando graves essas falhas, pois interrompem a rede por um longo tempo, o presente trabalho apresentou uma proposta de solução para o problema de interrupção em cadeia dos serviços da Rede Ufla: a implantação de redundâncias físicas e a posterior otimização desta solução pelo *Spanning Tree Protocol*.

O STP é um importante protocolo que deve ser implantado em toda rede computacional a fim de otimizar seu tráfego e aumentar seu tempo de operação (*Up Time*). Outros projetos de grande importância para a melhora das condições da rede da Universidade já foram implantados, como é o caso da segmentação lógica usando VLANS, desenvolvido por outros alunos do curso de Ciência da Computação desta Universidade.

Oriundos da arquitetura de redes de meio compartilhado, os pacotes de difusão (*broadcast*) são a principal causa da sobrecarga dos equipamentos de interconexão. Em redes com redundância, eles prejudicam largamente o desempenho desta, pelo fato de que os pacotes trafegam entre os enlaces redundantes indefinidamente.

Para definir a melhor solução a ser adotada, foi realizada uma série de etapas para a pesquisa, análise, teste e viabilização da solução.

Este trabalho trouxe grande contribuição por encontrar uma solução para problemas críticos da Rede Ufla, compatível com a disponibilidade financeira de uma instituição pública como a Ufla e que, de certa forma, se adapta à atual estrutura da rede.

REFERÊNCIAS BIBLIOGRÁFICAS

(BOYLES et. al.,2001) BOYLES, Tim; HUCABY, Dave. ***Cisco CCNP Switching Exam Certification Guide***. Cisco Press, 2001.

(COMER, 1999) COMER, Douglas. ***Computer Networks and Internets*** – 2 ed. Prentice Hall Inc, 1999.

(DIÓGENES, 2002) DIÓGENES, Yuri. ***Certificação Cisco – CCNA 3.0 Guia de Certificação para o Exame #640-607 –*** 2002, 2ª Edição, Axcel Books do Brasil Editora Ltda.

(FEIBEL, 1996) FEIBEL, Werner. ***Encyclopedia of Networking.*** SYBEX Inc, 1996.

(FURTADO, 2003) FURTADO, Leonardo. ***Switching: Spanning Tree Protocol (STP).*** CiscoTrainingBr.com. Disponível em: http://www.ciscotrainingbr.com. Acesso em 30/11/2004.

(FURUKAWA, 2003) FURUKAWA. ***Data Cabling System***. Guia Didático. Curso de Cabeamento Estruturado. Curitiba, 2003.

(JACK, 2003) JACK, Terry. ***CCNP: Building Cisco Multilayer Switched Networks.*** SYBEX Inc, 2003.

(KUROSE & ROSS, 2003) KUROSE, James F.; Keith W. ROSS. ***Rede de computadores e a Internet: uma nova abordagem***; Tradução Arlete Simille Marques; revisão técnica Wagner Luiz Zucchi – 1ª Edição – São Paulo : Addison Wesley, 2003.

(ODOM, 2004) ODOM, Wendell. ***Cisco CCNA ICND Exam Certification Guide***. Cisco Press, 2004.

(PINHEIRO, 2004) PINHEIRO, José Maurício Santos. ***Conceitos de Redundância e Contingência.*** Projeto de Redes. Disponível em: http://www.projetoderedes.com.br. Acesso em 30/05/2005.

(SOARES et. al., 1995) SOARES, Luiz Fernando Gomes; Guido Lemos; Sérgio Colcher. ***Redes de computadores***: das LANs, MANs e WANs às redes ATM; 2ª Edição – Rio de Janeiro : Campus, 1995.

(STEVENS, 1993) STEVENS, W. Richard. ***TCP/IP Illustrated Vol.1 – Protocols***. Addison-Wesley, 1993.

(TANENBAUM, 1997) TANENBAUM, Andrew S.. ***Redes de Computadores***: tradução [ds 3. ed. original] Insight Serviços de Informática. Rio de Janeiro: Campus, 1997.

(TORRES, 2001) TORRES, Gabriel. ***Redes de Computadores Curso Completo –*** 2001, Axcel Books do Brasil Editora Ltda.

(THIOLLENT, 1997) THIOLLENT, Michel. **Pesquisa-ação nas organizações**. São Paulo: Atlas, 1997.

(TRIVIÑOS, 1987) TRIVIÑOS, A. N. S. **Introdução à pesquisa em ciências sociais**. São Paulo: Ed. Atlas, 1987.

RESUMO ESTENDIDO

Em tempos em que a competitividade faz com que as organizações preocupem-se cada vez mais com a racionalização e o aproveitamento máximo de seus recursos, a fim de obter ganhos de eficiência, é imprescindível a procura constante de novas soluções.

Dentro desta visão, pode-se lançar mão de recursos já disponíveis no mercado. Quando projetos de redes utilizam múltiplos *switches*, a maioria dos engenheiros de redes inclui segmentos redundantes entre os *switches*. O objetivo disso é simples: criar enlaces alternativos para o tráfego de dados. Um *switch* pode falhar, um cabo pode ser cortado ou desconectado, e se houver um enlace redundante na rede, o serviço ainda estará disponível para a maioria dos usuários.

Porém, projetos de redes com enlaces físicos redundantes podem fazer com que os dados nessa rede entrem em *loop* infinito, ou seja, permaneçam em tráfego constante, congestionando a rede e gerando problemas significativos de performance.

Para minimizar esses problemas pode-se implementar uma técnica de otimização de tráfego em redes chamada *Spanning Tree Protocol* (STP) para impedir que os dados trafeguem indefinidamente pelos enlaces redundantes. Este recurso acompanha a grande maioria dos *switches*, porém, é ignorado e até desprezado como recurso de aumento de performance pelo público comprador.

Diante disso, o presente trabalho elaborou uma proposta de otimização de tráfego de dados na rede Ufla, através da implementação da técnica de *Spanning Tree Protocol* utilizando equipamentos já disponíveis – os *switches*.

Foi feita uma análise minuciosa da Rede Ufla, identificando os principais problemas enfrentados tanto pelos administradores quanto pelos usuários, as adaptações necessárias para possibilitar a implementação do STP e os pontos de maior necessidade de melhoria. Um teste-piloto foi realizado em laboratório para validar os benefícios e a funcionalidade da técnica.

Os resultados obtidos contribuíram consideravelmente não só para a viabilidade do aumento de performance da rede implementando a técnica testada, mas para conhecer e documentar as características estruturais e peculiares da Rede Ufla.