Arquitetura TCP/IP

Redes de computadores, modelos RM-OSI e TCP/IP

Versão 2.3
Agosto de 2018

Prof. Jairo

jairo@uninove.br
professor@jairo.pro.br

http://www.jairo.pro.br/

Arquitetura TCP/IP

"Arquitetura TCP/IP" tem por única intenção reunir um conteúdo acadêmico necessário para auxiliar o ensino de Arquitetura de Redes nos cursos de Tecnologias em Redes de Computadores, Segurança da Informação, Análise e Desenvolvimento de Sistemas, Banco de Dados, Sistemas para Internet, etc.

O conteúdo destina-se a oferecer uma noção básica em Arquitetura de Redes, abordando RM-OSI e TCP/IP. Toda a ênfase do conteúdo é conceitual.

O conteúdo aqui exposto pode ser livremente redistribuído e usado como apoio de aula, desde que mantenha a sua integridade original.

O arquivo "arquitetura_tcp_ip.pdf" está em "http://www.jairo.pro.br/arq_tcp_ip_e_redes_de_comput/".

Qualquer crítica ou sugestão, favor entrar em contato com o Prof. Jairo no endereço eletrônico "jairo@uni9.pro.br" ou "professor@jairo.pro.br".

Prof. Jairo - jairo@uninove.br – http://www.jairo.pro.br/

São Paulo, 19 de agosto de 2018.

ARQUITETURA TCP/IP

Sumário

Capítulo 1 - Rede de Computadores

Numa definição simples, rede de computadores é formada por um conjunto de módulos processadores capazes de trocar informação e compartilhar recursos, interligados por um sistema de comunicação que faz uso de um protocolo de comunicação comum a todos esses módulos.

A rede serve como meio de comunicação para compartilhamento de informações com redução de custos e deve apresentar confiabilidade e escalabilidade. Numa rede, a informação trafega em forma de pacotes de dados.

A redes podem ser classificadas pela *área* ou pela *topologia*.

1.1 - Classificação pela área

A classificação plea área define a rede pelas suas dimenões e abrangência física, que são *LAN*, *MAN*, *WAN*, *WLAN*, *VPN*, *PAN*, *CAN*, *GAN*, *HAN* e *SAN*.

1.1.1 - LAN (Local Area Network)

São redes locais com abrangência de até aproximadamente 10 Km. A LAN tem três características distintas: tamanho limitado, tecnologia da transmissão e topologia.

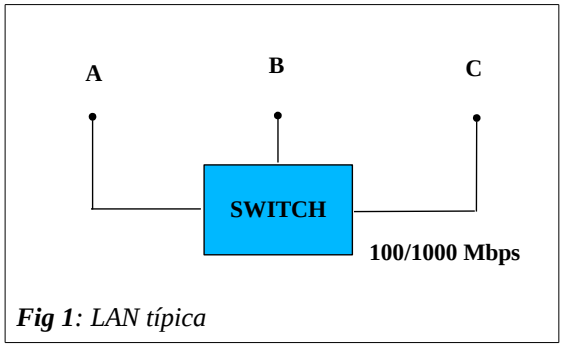


***Fig 1**: LAN típica*

1.1.2 - MAN (Metropolitan Area Network)

São redes com abrangência entre aproximadamente 10 a 100 Km.

1.1.3 - WAN (Wide Area Network)

São redes com abrangência numa área maior que 100 Km. O melhor exemplo é a rede pública de abrangência mundial conhecida como internet.

1.1.4 - WLAN (Wireless Local Area Network)

São redes locais que usam para comunicação tecnologias wireless, como o wi-fi.

1.1.5 - VPN (Virtual Private Network)

A VPN é uma rede privada criada dentro de uma rede pública (internet, WAN). Normalmente é usado VPN para, de casa, com o uso da internet, acessar o ambiente de trabalho. Este acesso é seguro, pois a VPN cria um túnel protegido dentro da rede pública.

1.1.6 - PAN (Personal Area Network)

São redes usadas em residências ou pequenos escritórios, atualmente se populariza devido à evolução da comunicação *wireless* (sem cabos) nesse tipo de rede, que facilita e barateia a instalação.

1.1.7 - CAN (Campus Area Network)

É uma rede MAN de alguma Universidade, com objetivo de interligar os campi espalhados numa região metropolitana. A rede da Uninove pode ser classificada como CAN.

1.1.8 - GAN (Global Area Network)

São redes usadas principalmente por multinacionais, que devido a sua extensão global necessita de uma rede privada de grandes extensões. Exemplo: McDonalds.

1.1.9 - HAN (Home Area Network)

É uma rede local doméstica, existe apenas numa residência. Normalmente usa wi-fi.

1.1.10 - SAN (Storage Area Network)

São redes usadas para interligar dispositivos de armazenamento de massa (HDs) externos (*storages*), dispositivos de backup (fitas) e servidores nos *data centers* (centro de dados).

1.2 - Classificação pela topologia

Na classificação pela topologia adotam-se os métodos de interconexão, que são *ponto a ponto* e *difusão*.

1.2.1 – Ponto a ponto

A rede é ponto a ponto quando a comunicação ocorre apenas em dois nós ligados fisicamente. As redes ponto a ponto pela topologia classificam-se em *estrela*, *laço* e *árvore*.

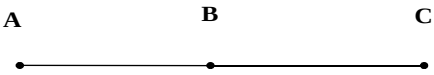


***Fig 2**: rede ponto-a-ponto típica*

1.2.1.1 - Estrela

A rede estrela normalmente usa um concentrador de rede (*switch*), cuja função é conectar dois nós. Embora essa topologia seja muito popular, tem como desvantagem um ponto único de falha, que ocorre caso o concentrador falhe.

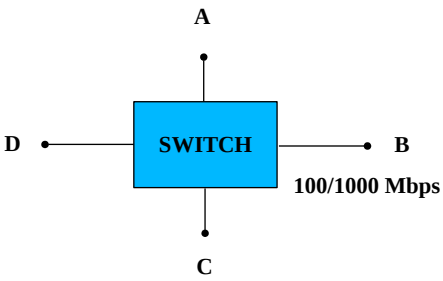


***Fig 3**: rede estrela*

1.2.1.2 - Laço

A topologia em laço é apenas uma versão modificada da estrela, porém nesse caso não existe a necessidade do *switch*.

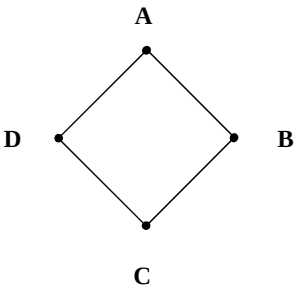


***Fig 4**: rede em laço*

1.2.1.3 - Árvore

A árvore é uma configuração hierárquica.

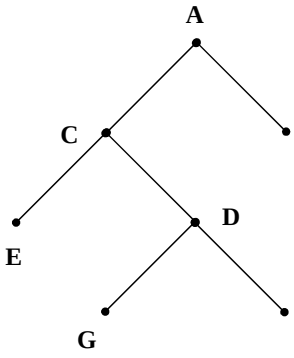


***Fig 5**: configuração hierárquica*

1.2.2 - Difusão

Nessa topologia de rede, os módulos processadores compartilham um canal de comunicação único e os dados enviados por um módulo são recebidos por todos os outros. Neste caso, é necessário algum método para controlar o acesso simultâneo a essa rede, ou seja, faz-se necessário alguma arbitragem.

Como exemplo, no caso da rede *estrela* (figura 3, acima), temos um anel lógico sobre uma estrela física, e por usar um switch é classificado como topologia ponto a ponto. Mas se substituir o switch por um hub, então torna-se topologia difusão.

No caso da estrela física com tecnologia Ethernet, costuma-se usar o algoritmo CSMA/CD[1] (Capítulo 6, IEEE802.3) como árbitro, e no laço simples usa-se normalmente o *token*. As redes de difusão classificam-se em *barramento* e *anel*.

1.2.2.1 - Barramento

É a topologia onde os nós na rede apresentam-se em forma de uma barra, como exemplo temos as antigas redes *ethernet* que usavam cabos coaxiais.

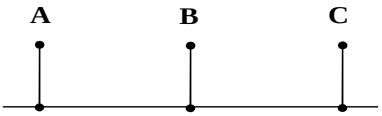


***Fig. 6**: barramento*

1.2.2.2 - Anel

Nessa topologia todos os nós estão conectados no mesmo anel. Na figura 7 temos um laço simples, que é um anel lógico sobre um anel físico.

Uma observação importante é que nesse caso o meio é compartilhado, mas do ponto de vista físico as comunicações são ponto a ponto.

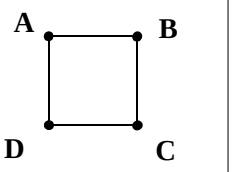


***Fig 7**: anel*

1 CSMA/CD protocol: Carrier Sense Multiple Access with Collision Detection.

Capítulo 2 - Endereçamento

A definição de endereçamento envolve alocar um endereço para cada nó da rede, endereço esse que seja unívoco dentro dessa mesma rede.

O endereço pode ser tanto físico quanto lógico. Se estamos em uma rede local (LAN), a comunicação entre os nós (módulos processadores) é feita usando-se o endereço físico, se a comunicação envolve o inter redes (internet), é necessário um endereço lógico.

No caso do endereço físico temos como exemplo o MacAdress, media access control address (*Ethernet*), que é composto de 6 bytes (48 bits), por exemplo 44:8a:5b:94:63:9a. Ainda como exemplo, apesar de haver um número muito grande de fabricantes de interfaces de redes, não existe o caso de dois desses equipamentos terem o mesmo endereço Mac pois os três primeiros bytes do endereço físico é um número constante determinado para aquele fabricante específico[2], e a outra parte do endereço é o próprio fabricante que determina, no estilo *serial number* (número de série).

No caso do endereço lógico temos como exemplo o IP da arquitetura internet (TCP/IP, *Transmission Control Protocol/Internet Protocol*). A versão mais usada atualmente do TCP/IP é a versão 4, ou ipv4.

A representação mais usada para o endereço ipv4 é:

xxx.xxx.xxx.xxx

Onde xxx é um número decimal entre 0 e 255. Como exemplo de endereço ipv4 temos 192.168.1.10, que é um endereço de 32 bits (4 bytes).

Porém, aos poucos está chegando a nova versão do TCP/IP, que é a versão 6 ou ipv6. A representação mais comum para ipv6 é:

hhhh:hhhh:hhhh:hhhh:hhhh:hhhh:hhhh:hhhh

Onde hhhh é um número hexadecimal de 4 dígitos (16 bits ou 2 bytes), por exemplo fe80. Como exemplo de endereço ipv6 temos fe80::468a:5bff:fe94:639a, que é um endereço de 128 bits (16 bytes).

A razão dessa gradual substituição de endereços ipv4 por ipv6 é devido ao esgotamento da disponibilidade de endereços para atender a uma demanda que aumenta ano após ano.

Abaixo é mostrado o número máximo (teórico) de endereços nas versões ipv4 e ipv6.

2 OUI: *Organizationally Unique Identifier* (Identificador Único de Organização) é um número de 24 bits que identifica univocamente um vendedor, fabricante ou outra organização que produza ou distribua equipamentos ou dispositivos que contenham endereços físicos.

ipv4: 2^{32} => 4294967296 ($4{,}3x10^{9}$)
ipv6: 2^{128} => $3{,}4x10^{38}$

Portanto, com a adoção do ipv6, haverá um incremento em cerca de 10^{29} vezes mais endereços disponíveis. Isto provavelmente deverá atender a demanda de endereços que a *internet das coisas* irá necessitar nas próximas décadas.

Outro aspecto importante a ser considerado é que na versão ipv6 o endereço físico (MacAddress) é incorporado ao endereço lógico, resultando numa construção redundante. Convém lembrar que no ipv4 o endereço lógico e o endereço físico são distintos para uma mesma interface.

E como o endereço físico é de 48 bits, então o número máximo (teórico) de endereços é:

MacAddress: 2^{48} => $2{,}8x10^{14}$

Portanto, o número máximo de MacAddress é cerca de 6500 vezes maior que o de endereços ipv4. E se os endereços ipv4 já estão esgotados, não vai levar muito tempo para esgotar também os endereços físicos algum tempo após a chegada da internet das coisas.

Convém notar que o OUI consome metade do espaço de endereçamento do MacAddress, portanto na prática o número máximo de endereços possíveis é bem menor que $2{,}8x10^{14}$. Supondo que haja mil (10^{3}) organizações, então o número máximo possível de endereços será:

$10^{3} \times 10^{24} = 10^{3} \times 16{,}8 \times 10^{6} = 16{,}8 \times 10^{9}$

Outro problema é que o endereço físico, ao contrário do endereço lógico, não é reciclável. Por exemplo, uma interface de rede fabricada há 15 anos pode já ter sido descartada, como pode muito bem ainda estar em uso.

O IEEE[3] usa a notação EUI-48 para designar o atual MacAddress. EUI-48 significa *Extended Unique Identifier* (Identificador Único Estendido). Em vista do espaço de endereçamento limitado do EUI-48, o IEEE encoraja a adoção do EUI-64, que estende de 48 para 64 bits o tamanho do MacAddress.

3 IEEE - *Institute of Electrical and Electronics Engineers* (Instituto de Engenheiros Eletricistas e Eletrônicos).

Capítulo 3 - Roteamento

A função do roteamento consiste no processo de escolha do melhor caminho que um pacote de dados tome ao viajar entre os nós de origem e de destino.

Quando os nós processadores estão na mesma sub-rede (mesma LAN), a tarefa de roteamento é trivial, porém quando os nós estão em sub-redes diferentes essa comunicação ocorre via *gateway*.

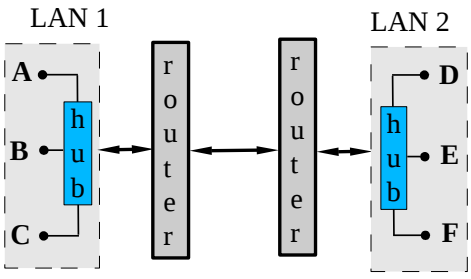


***Fig. 8**: roteadores ligando duas redes locais*

O *gateway* é que faz o roteamento dos pacotes de dados de uma sub-rede para outra, baseado em endereços lógicos de destino e origem.

Sub-rede, nesse caso, é uma rede vista do inter rede (internet) e não uma segmentação de rede (sub rede, Item 7.6.2.2).

Os roteadores são equipamentos projetados para executar a tarefa de roteamento, e possuem algoritmo interno que permite a realização dessa tarefa. No processo de roteamento o "melhor" caminho para o pacote é função da métrica (distância, quantidade de pulos e largura de banda).

O roteamento é a principal função de um gateway, e pode ser estático ou dinâmico.

3.1 – Roteamento estático

Se as rotas forem simples, elas podem ser configuradas estáticas, neste caso a tabela de roteamento é construída manualmente pelo administrador da rede.

As tabelas de roteamento estáticas não se ajustam automaticamente em acordo com as possíveis alterações na métrica da rede, desse modo roteamento estático deve ser utilizado somente onde as rotas não sofram alterações.

As vantagens do roteamento estático são a segurança proporcionada pela não divulgação de rotas usadas e à redução da sobrecarga na rede introduzida pela troca de pacotes de roteamento originadas pelo uso de protocolos de roteamento, que ocorre no roteamento dinâmico.

3.2 – Roteamento dinâmico

Nas redes onde existe muitas alternativas de rota para um mesmo ponto deve ser

utilizado roteamento dinâmico.

O roteamento dinâmico faz uso de protocolos de roteamento tais como RIP (Routing Information Protocol), OSPF (Open Short Path First), EIGRP (Enhanced Interior Gateway Routing Protocol) ou BGP (Border Gateway Protocol) .

Neste caso, a tabela de roteamento dinâmica é criada e mantida a partir de informações trocadas entre os roteadores pelos protocolos de roteamento. Estes protocolos são desenvolvidos para distribuir informações que ajustam as rotas dinamicamente de modo a refletir as alterações nas condições da rede.

A principal função de um protocolo de roteamento é fornecer as informações necessárias para poder fazer o roteamento, mediante criação e atualização de tabelas de roteamento.

Estes protocolos são desenvolvidos para alterar para uma rota alternativa quando a rota primária se tornar inoperável, além de decidir qual é a rota preferida para um destino.

Capítulo 4 - Confiabilidade da rede

4.1 - Detecção de erros

A detecção de erros se refere à integridade dos dados, ou seja, os dados recebidos devem ser idênticos aos que foram transmitidos. A detecção de erros pode ser implementada tanto pela *checagem de paridade* quanto *redundância cíclica*.

4.1.1 - Checagem de paridade

Checagem de paridade é uma forma simples de detecção de erros que consiste em determinar se os *bits* que foram transmitidos são os mesmos que foram recebidos pelo destinatário.

Para se determinar se houve introdução de ruído nessa transmissão (ou seja, algum *bit* de informação sofreu alteração do estado original) é gasto um *bit* dessa transmissão para guardar a informação de paridade, daí o nome *bit de paridade*.

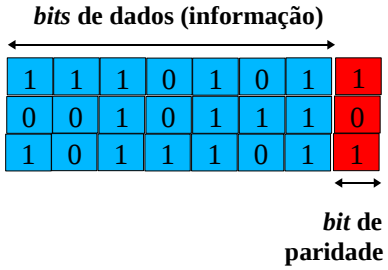


***Fig. 9**: exemplos de contagem de bits de paridade*

Num caso bem simples, pode-se imaginar a transmissão de um conjunto de dados de 7 *bits*, nesse caso o 8º *bit* será o de paridade. Para determinar o *bit* de paridade basta somar os 7 *bits* de dados e guardar esse valor no *bit* de paridade. O receptor, ao receber esses dados, fará esse mesmo cálculo, e caso a paridade concorde, pode-se admitir que os dados estão íntegros.

A maior vantagem dessa forma de detecção de erros é a simplicidade e consequente economia de recursos computacionais, porém ela falha caso mais de um *bit* de dados seja alterado. Falha também caso o próprio *bit* de paridade seja alterado na transmissão.

Nota: na contagem de *bits* de paridade, 1+1=0 pois estamos considerandos dois dígitos binários (duas casas), e então a conta será 01+01=10 que é 2 na base binária. Já 0+0=0, 1+0=1 e 0+1=1.

4.1.2 - Redundância cíclica (CRC)

O CRC (*Cyclic Redundancy Code*) é uma forma de detecção de erros mais sofisticada e robusta que o *bit* de paridade e usa um polinômio de grau *n*-1 para *n bits* de dados trafegados. Para checar uma série de *bits*, CRC constrói um polinômio algébrico cujos coeficientes dos termos são os valores dos *bits* nessa série. Por exemplo, vamos considerar o seguinte conjunto de dados composto de 7 *bits*: 1011010.

Nesse caso, o polinômio é $1x^6 + 0x^5 + 1x^4 + 1x^3 + 0x^2 + 1x + 0$,

que se reduz a $x^6 + x^4 + x^3 + x$.

A seguir, esse polinômio é dividido por um *polinômio gerador predeterminado* (CRC-16, CRC-32, etc.), e o conjunto de dados fica sendo o dividendo e o polinômio gerador é o divisor. O resto dessa divisão é a checagem de soma CRC, que deve ser incluída no quadro. O nó processador efetua um cálculo análogo ao do nó emissor, e se o resto dessa divisão for zero, o quadro é considerado íntegro.

A eficiência do CRC é função do polinômio gerador usado, CRC-16 detecta 100% de erros únicos e duplos, já CRC-32 é tão mais rigoroso que as chances de termos dados ruins recebidos e não detectados é da ordem de 1 em 4.3 bilhões ($2^{23} – 1$).

Por outro lado, quanto mais sofisticada essa eficiência, maior o custo em processamento para efetuar esses cálculos.

Capítulo 5 - Arquitetura de Rede

5.1 - Definição

Quanto à arquitetura, a rede divide a tarefa de comunicação em várias camadas funcionais, onde a camada inferior presta serviços à camada superior que requisita esses serviços.

Como exemplo, imaginemos o caso de um diretor de um setor de uma empresa (empresa A) que queira enviar um documento a outro diretor em outra empresa (empresa B). A maneira convencional é esse diretor transferir a tarefa para a sua secretária, que por sua vez redige o documento e envia para o boy. O boy, por sua vez, entrega o documento ao chefe de malote que despacha o documento para o endereço correto. Uma vez chegando lá, o documento segue todo esse cerimonial na "pilha" hierárquica, porém agora em sentido inverso, até chegar às mãos do diretor. Embora burocrático, esse procedimento traz vantagens, a maior de todas é liberar os diretores das tarefas mais básicas as quais são atribuídas aos seus subordinados.

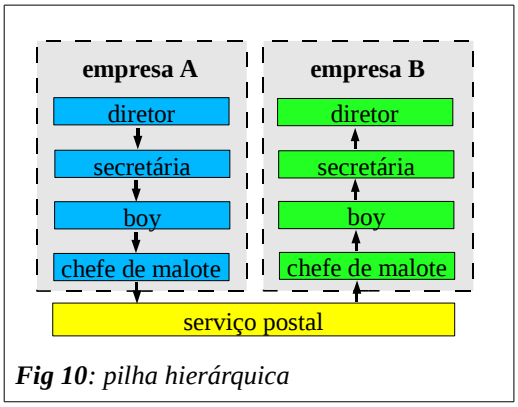


***Fig 10**: pilha hierárquica*

No caso de uma arquitetura de rede, no lugar do diretor está o *software* aplicativo do usuário, por exemplo um navegador da internet, e no caso do serviço postal está o meio físico de comunicação em rede, como exemplo temos o cabo de rede.

5.2 - Histórico

A arquitetura de rede amplamente utilizada hoje é o TCP/IP (*Transmission Control Protocol/Internet Protocol*), que teve sua origem em meados da década de 1960 como um projeto militar do Departamento de Defesa dos EUA e que foi desenvolvido na ARPA (*Advanced Research Project Agency*). Desse projeto resultou uma rede inicialmente conhecida como ARPANET, que entrou em operação experimental em 1969 e que posteriormente introduziu novidades no conceito de comunicação.

Entre essa novidades estão *comutação de pacotes* (conceito de roteamento de pacotes), divisão da tarefa de comunicação em *camadas funcionais* (conceito de arquitetura de rede) e interligação de computadores entre universidades americanas e de outros países.

Simultaneamente a isso, outros fabricantes já possuíam as suas arquiteturas proprietárias, como era o caso da IBM com SNA (*Systems Network Architecture*) e Digital com Decnet.

No final dos anos 1970 havia uma demanda potencial de crescimento para redes, porém também havia uma crise criada pela heterogeneidade de padrões, protocolos e equipamentos de comunicação. Por exemplo, ARPANET com arquitetura específica para atender as suas redes.

Tudo isso levou a um esforço para o desenvolvimento e implantação de arquiteturas abertas, e é nesse contexto que surge o modelo de referência RM-OSI (*Reference Model/Open Systems Interconnect*).

Sabemos hoje que o RM-OSI, desde a sua criação no início dos anos 1980, manteve-se na prática apenas como um modelo acadêmico ou modelo padrão, enquanto o TCP/IP - por ser aberto, simples e o primeiro a se difundir pelas redes do mundo inteiro - se tornou o modelo *de fato* usado no mercado.

5.3 - Solução ISO (International Organization for Standardization)

Entre 1978 e 1982 foi apresentada a solução ISO, definindo um modelo de referência (RM-OSI) para interconexão de sistemas abertos.

RM-OSI são as iniciais de *Reference Model – Open Systems Interconnections*, que significa Modelo de Referência para Interconexão de Sistemas Abertos.

A arquitetura OSI é um grande projeto de engenharia de protocolos, e está dividida em 7 camadas funcionais: *aplicação*, *apresentação*, *sessão*, *transporte*, *rede*, *enlace* e *física*.

***Fig 11**: arquitetura RM-OSI*

5.4 - TCP/IP – Internet

Esta arquitetura foi lançada pelo Departamento de Defesa do governo americano e hoje é o padrão *de facto* do mercado.

Todas as definições dessa arquitetura estão em RFC (*Request for Comments*) elaboradas pelo IAB (*Internet Activities Board*), ou seja, também é um padrão aberto.

O TCP/IP é composta por dois protocolos principais, o IP (*Internet Protocol*) e o TCP (*Transmission Control Protocol*). O endereçamento IP é do tipo datagrama (não orientado à conexão), já o TCP é protocolo de transmissão (transporte) e é orientado à conexão.

O TCP/IP oferece um serviço relativamente confiável, mesmo em redes não confiáveis. Em redes de alta qualidade, onde a confiabilidade não é importante, pode-se utilizar o UDP (*User Datagram Protocol*) que não é orientado à conexão.

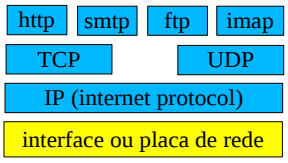


***Fig 12**: pilha TCP/IP com quatro camadas*

Capítulo 6 - O RM/OSI e as redes locais

6.1 - Redes Locais

Na implantação de uma rede local, a escolha de determinado tipo de rede e de equipamentos deve levar em consideração o conjunto de aplicações que rodarão lá. Além disso, outros fatores não podem ser esquecidos, que são o custo, tempo de resposta (10 Mbps, 100Mbps, 1Gbps), compatibilidade, etc.

As características das redes locais afetam os níveis mais baixos de protocolos da arquitetura de rede. Essas características são: elevado desempenho, baixo retardo, baixa taxa de erro, difusão, aplicações típicas de redes locais. O RM/OSI foi pensado para redes geograficamente distribuídas, portanto não confiáveis.

A aplicabilidade do RM/OSI em redes locais precisa levar em consideração essas características das redes locais.

6.2 - O padrão IEEE802

Com as características que as LAN´s têm, não cabe ao nível de enlace utilizar muitos *bits* de redundância para recuperação de erros. Nas LAN´s, os protocolos de ligação entre nós poderiam estar tanto no nível 2 (enlace) como no nível 3 (rede), e existem propostas para colocar esse protocolo de ligação no nível 1 (físico).

Além disso, nas LANs normalmente existem padrões diferentes, com por exemplo *Ethernet*, *Token Ring*, etc.

Para tentar minimizar esses problemas, a Sociedade de Computação do Instituto de Engenharia Eletricista e Eletrônica dos EUA (*IEEE Computer Society*) criou o IEEE 802, também conhecido por IEEE 802 LAN/MAN Standards Committee (www.ieee802.org).

Os objetivos do IEEE 802 são obter uma arquitetura padrão, que seja orientada para o desenvolvimento de redes locais e com as seguintes características:

- Correspondência máxima com o RM/OSI;
- Interconexão eficiente de equipamentos a custo moderado;
- Implantação da arquitetura a custo moderado.

O IEEE 802 pode ser visto como uma adaptação das duas camadas inferiores da RM/OSI.

No IEEE 802 existem 3 camadas, sendo uma equivalente à camada física e outra equivalente à camada de enlace. Essas camadas são assim denominadas:

- Camada Física (PHY)
- Sub camada de controle de acesso ao meio (MAC)
- Sub camada de controle de enlace lógico (LLC)

O IEEE802 tem várias subdivisões, quatro delas merecem ser mencionadas: 802.3, 802.5, 802.4 e 802.11.

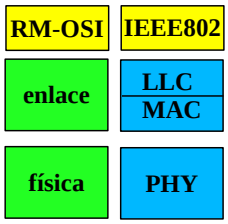


***Fig 13**: comparação entre RM-OSI e IEEE802*

6.2.1 - IEEE802.3

O IEEE802.3 trata da tecnologia Ethernet, tanto ponto a ponto (full-duplex) quanto com CSMA/CD (*Carrier Sense Multiple Access with Collision Detection*), onde cada módulo processador nessa rede "escuta" antes de transmitir, ou seja, cada nó somente tenta transmitir quando ninguém está transmitindo, com isso reduz a ocorrência de colisão de pacotes (mais de um dispositivo tenta transmitir ao mesmo tempo). Convém notar que precisa haver alguma cooperação e arbitragem entre esses módulos, caso contrário nenhum conseguirá transmitir.

6.2.2 - IEEE802.5

O IEEE802.5 trata da tecnologia Token Ring (IBM/1970), que tem como método de arbitragem de acesso ao meio a passagem de um pequeno quadro, chamado *token*: apenas o nó na rede que estiver de posse do *token* é que pode transmitir, na sequência esse nó transmite o *token* e perde o direto de transmitir até recebê-lo de volta. O *token* circula no anel.

6.2.3 - IEEE802.4

O IEEE802.4 trata da tecnologia FDDI, Fiber Distributed Data Interface, que geralmente usa fibra ótica e pode alcançar centenas de quilômetros. As fibras são usadas tanto em LAN quanto em MAN e WAN. São muito usadas em SAN.

6.2.4 - IEEE802.11

O IEEE802.11 trata das tecnologias wireless em LAN, que é o WLAN.

Capítulo 7 - As camadas RM-OSI e TCP/IP

7.1 - Visão geral do RM-OSI

Podemos dividir o RM-OSI em 3 partes, uma parte inferior orientada a redes ou comunicação composta por 3 camadas (física, enlace e rede), uma parte central que verifica a entrega dos dados composta pela camada de transporte e uma parte superior orientada a aplicações ou serviços composta pelas camadas de sessão, apresentação e aplicação.

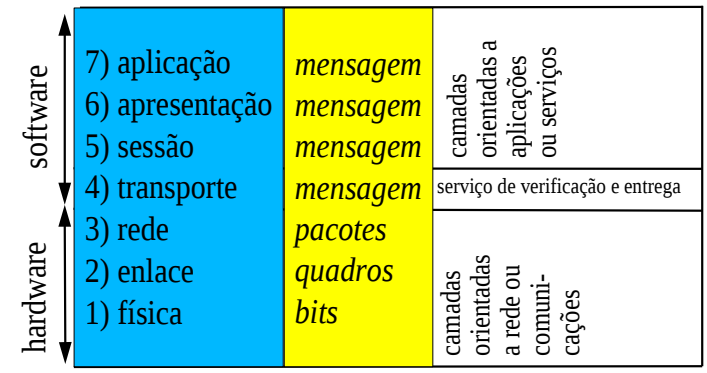


***Fig 14**: Visão geral das camadas do RM-OSI*

7.2 - Visão geral das 7 camadas do RM-OSI

O RM-OSI está dividido em 7 camadas: *física*, *enlace*, *rede*, *transporte*, *sessão*, *apresentação* e *aplicação*.

7.2.1 - Física

É a camada responsável por transmitir *bits* através de uma ligação. Aceita quadros da camada de enlace de dados e traduz esses *bits* em sinais do meio físico. Cuida de questões como o tipo do cabo em uso e o esquema de sinalização. Define o modo de transmissão (unidirecional, bidirecional, etc), modo de conexão (ponto a ponto, multiponto) e modo de tratamento dos erros.

7.2.2 - Enlace

É a camada responsável pela transferência de dados entre pontos de uma ligação física, fraciona as mensagens recebidas do emissor em unidades de dados denominadas quadros, que correspondem a algumas centenas de *bytes*. Essa camada trata de detecção de erros e controle de fluxo, que é a necessidade de armazenamento de dados a transmitir quando a transmissão não for efetuada a uma mesma taxa (por exemplo, 100 e 1000 Mbps). Essa camada também resolve problemas relativos a quadros danificados, perdidos ou duplicados.

7.2.3 - Rede

Nessa camada, as mensagens formatadas são denominadas pacotes. É função dessa camada encaminhar os pacotes de dados do emissor ao receptor. Essa camada deve resolver todos os problemas relacionados à interconexão de redes heterogêneas, como por exemplo incompatibilidades no endereçamento e incoerências com relação ao tamanho das mensagens.

7.2.4 - Transporte

A função dessa camada é aceitar dados da camada de sessão, quebrar esses dados em pacotes menores se necessário e passá-los para a camada de rede. Uma característica dessa camada é implementar um diálogo fim a fim, ou seja, o processo executando no sistema fonte dialoga com o processo executando no nó destino através de cabeçalhos (*headers*) e informações de controle contidas nas mensagens desse nível. Essa camada implementa um mecanismo de controle de fluxo fim a fim para evitar que o sistema fonte envie mensagens numa taxa superior àquela que o sistema destino possa receber. Normalmente cria uma conexão de rede para cada conexão de transporte requerida pela camada de sessão.

7.2.5 - Sessão

Essa camada trata da coordenação entre processos de comunicação entre os nós na rede, verifica se uma conexão permite comunicação em duplex parcial ou completo, sincroniza fluxo de dados e restabelece conexão em caso de falha.

NOTA:
Os serviços em rede podem ser *orientados à conexão*, como é o caso do SAP (*service access point*) ou socket (Unix) ou podem ser *sem conexão*.

7.2.6 - Apresentação

Essa camada trata de formato de dados, traduções e conversões de código. Cuida da sintaxe e semântica dos dados transmitidos, além da compressão e criptografia. Na prática, essa camada é frequentemente incorporada na camada de aplicação.

7.2.7 - Aplicação

Essa camada consiste de protocolos que definem aplicações específicas orientadas para usuários, como correio eletrônico e transferência de arquivos.

7.3 - Visão geral do TCP/IP

O termo TCP/IP designa uma família de protocolos de comunicação de dados, tais como FTP, SMTP e HTTP.

Essa família de protocolos teve origem na ARPANET, um projeto do Departamento de Defesa dos EUA. Essa família de protocolos foi desenvolvida para ser usada num meio não-confiável, mas mesmo assim atualmente o TCP/IP é amplamente usado até em LANs que não têm acesso à internet.

O segredo do sucesso do TCP/IP vem principalmente do fato dele ter sido o primeiro protocolo de comunicação em rede a atingir uma abrangência mundial.

Outras características igualmente importantes do TCP/IP são:

- protocolo aberto, público, independente de equipamentos e Sistemas Operacionais;
- não define protocolo para o nível físico, podendo por exemplo usar *ethernet* e *token ring*;
- esquema de endereçamento unívoco;
- protocolos de aplicação que atendem à demanda dos usuários.

7.4 - Visão geral das 4 (ou 5) camadas do TCP/IP

Ao contrário do modelo RM-OSI, que tem um compromisso acadêmico de ser um modelo de referência, a arquitetura do protocolo TCP/IP em camadas não tem qualquer outro compromisso que não seja a funcionalidade.

Desse modo, estabelecer relação precisa entre as camadas do modelo RM-OSI e TCP/IP torna-se tarefa difícil.

O modelo mais aceito é dividir a arquitetura TCP/IP em 4 camadas: acesso à rede ou interface de rede, internet, transporte e aplicação.

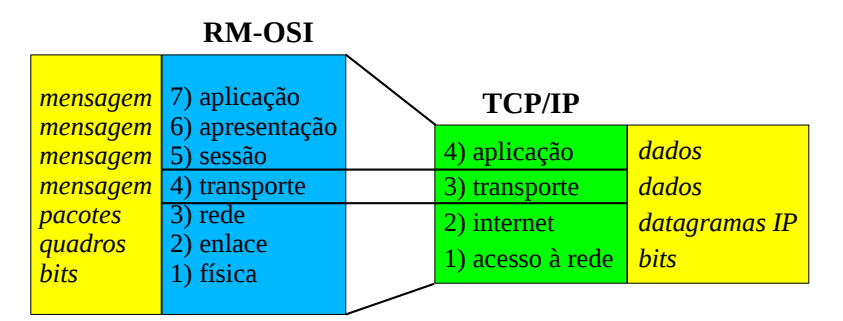


Fig. 15: *arquitetura TCP/IP com quatro camadas comparada com RM-OSI*

Mas também é comum apresentar o TCP/IP em 5 camadas, que é o modelo TCP/IP híbrido: camadas física, enlace de dados, internet, transporte e aplicação.

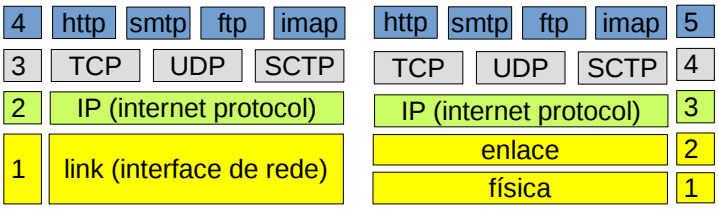


Fig. 16: *arquitetura TCP/IP com quatro e com cinco camadas*

Semelhante ao modelo RM-OSI, cada camada da pilha de protocolos adiciona um cabeçalho (*header*) com informações de controle. Essa adição de informações de controle é denominada *encapsulamento*.

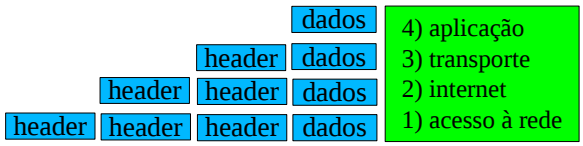


Fig. 17: *encapsulamento na pilha TCP/IP*

7.4.1 - Acesso à rede

Essa camada provê os meios para que os dados sejam transmitidos a outros nós processadores na mesma rede. Essa camada pode abranger as 3 primeiras camadas do RM-OSI, porém não define propriamente os protocolos para esses 3 níveis e sim como utilizar os protocolos já existentes para suportar a transmissão.

7.4.2 - Internet

A camada internet tem como principais funções:

- definir o datagrama IP, que é a unidade básica de transmissão;
- definir o esquema de endereçamento IP;
- rotear datagramas IP;
- fragmentar e remontar datagramas IP.

7.4.3 - Transporte

Os principais protocolos dessa camada são TCP e UDP. O TCP é orientado à conexão com detecção e correção de erros fim a fim, já o UDP é não orientado à conexão e não confiável, por outro lado o UDP é muito leve (causa pouco *overhead*) na rede.

Como exemplo comparativo, o TCP tem aspecto de ligação telefônica (completa a ligação, que é chamada de *conexão*) enquanto o UDP se assemelha ao serviço postal (correio), no sentido de transmitir pacotes isolados.

7.4.4 - Aplicação

A camada aplicação é a que provê protocolos que se comunicam, de um lado, com os aplicativos do usuário (lado cliente) e na outra ponta com os aplicativos servidores (serviços).

Exemplos de protocolos da camada de aplicação: telnet, FTP, SMTP, DNS.

<table>
<tr><th>RM-OSI</th><th colspan="3">TCP/IP</th></tr>
<tr><td>7) Aplicação</td><td rowspan="3">SNMP, TFTP, NFS, DNS, BOOTP</td><td rowspan="3">FTP, TELNET, FINGER, SMTP, POP, IMAP, SSH, HTTP</td><td rowspan="3">Aplicação</td></tr>
<tr><td>6) Apresentação</td></tr>
<tr><td>5) Sessão</td></tr>
<tr><td>4) Transporte</td><td>UDP</td><td>TCP</td><td>Transporte</td></tr>
<tr><td>3) Rede</td><td colspan="2">IP, icmp[2]</td><td>Internet</td></tr>
<tr><td>2) Enlace</td><td colspan="2">placas de interface de rede</td><td rowspan="2">Interface de rede</td></tr>
<tr><td>1) Física</td><td colspan="2">meio de transmissão</td></tr>
</table>

Tabela 1: *arquitetura TCP/IP*

7.5 - As camadas do RM-OSI

7.5.1 - Camada física

A camada física é o *suporte de transmissão* que assegura o transporte de dados entre dois equipamentos terminais. Nesse caso, os dados são representados por um conjunto de *bits*.

Os suportes de transmissão classificam-se pela *existência de guia físico* (por exemplo, fibra ótica) e pela *ausência de guia físico* (por exemplo, ondas eletromagnéticas ou *wireless*).

7.5.1.1 - Suportes de transmissão com guia físico

Como exemplos de suportes de transmissão com guia físico temos o *par de fio trançado*, *cabo coaxial* e *fibra ótica*.

O *par de fio trançado* é o mais clássico de todos os meios de transmissão. Consiste de dois fios de cobre isolados e arranjados na forma helicoidal na longitude, isto é, enrolados um no outro. A técnica de enrolar o par é para diminuir os efeitos das induções eletromagnéticas parasitas, provenientes do ambiente.

2 icmp: *Internet Control Message Protocol*, um protocolo específico para diagnóstico de comunicação em rede.

O uso mais comum desse suporte de transmissão está na rede telefônica, e nesse caso o sinal pode percorrer alguns quilômetros sem necessidade de amplificação ou regeneração do sinal.

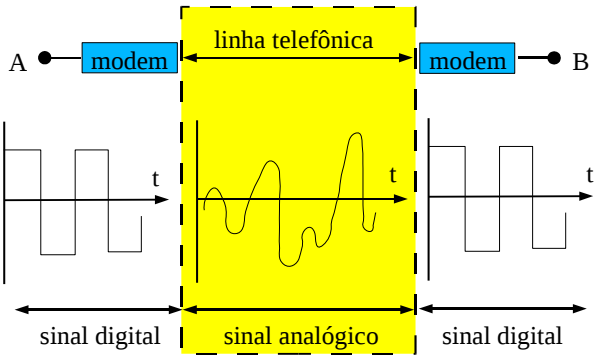


Fig 18: *modem modula e demodula o sinal*

O par de fio trançado pode ser usado tanto na transmissão de sinais analógicos (como é o caso do telefone) quanto digitais (telégrafo, computador). No caso da linha telefônica, para permitir que dois equipamentos terminais se comuniquem é necessário um *modem* em cada ponta dessa linha, pois os sinais em rede são binários e na telefonia são analógicos, daí a necessidade de uma "tradução" chamada de *modulação* e *demodulação* (MODEM).

A banda passante é função do diâmetro e pureza do condutor, além da natureza do isolante e comprimento do cabo. A taxa de transmissão está na faixa de algumas dezenas de Kbps (10^3 *bits* por segundo).

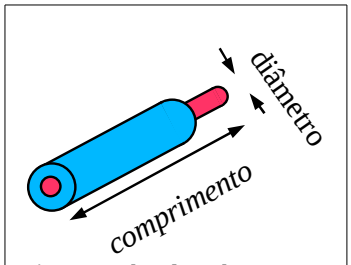


Fig 19: *cabo de cobre isolado*

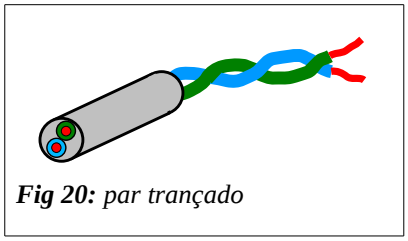


Fig 20: *par trançado*

Esse suporte de transmissão apresenta um baixo custo e grande faixa de utilização, por isso é dos mais utilizados atualmente.

No caso das LANs, costuma-se usar cabos com pelo menos 2 pares, onde um par é para transmitir e outro para receber os dados. Se esse conjunto for envolvido por uma capa simples, será do tipo UTP (*Unshielded Twisted Pair*), se tiver uma blindagem externa será STP (*Shielded Twisted Pair*).

Os cabos UTP estão divididos em categorias que têm a ver com a frequência de transmissão e que definem a taxa de transferência nesse meio. Por exemplo, a categoria 3 transmite a 10 MHz e pode ser usada para Ethernet (10 Mbps), já a categoria 5 transmite a 100 MHz e é adequada para FastEthernet (100 Mbps). É possível transmitir a 1 Gbps com categoria 5 se forem usados todos os 4 pares do cabo no esquema bidirecional, e não apenas dois pares como é o usual.

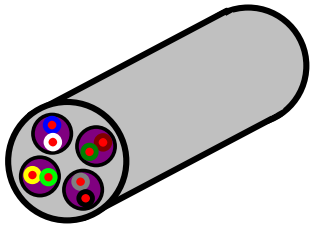


Fig 21: *corte em cabo UTP*

A tabela 2, abaixo, mostra um comparativo entre as diferentes categorias de cabaos UTP.

Categoria UTP	Taxa de transferência máxima	Comprimento máximo
CAT1	1 Mbps	-
CAT2	4 Mbps	-
CAT3	10 Mbps	100 metros
CAT4	16 Mbps	100 metros
CAT5	100 Mbps	100 metros
CAT5e	1 Gbps	100 metros
CAT6	10 Gbps	55 metros
CAT6a	10 Gbps	100 metros
CAT7	10 Gbps	100 metros (40 Gbps a 50 metros)

Tabela 2: *categorias de cabos UTP*

Os *cabos coaxiais* são constituídos por dois condutores arranjados de forma concêntrica: um condutor central, que é envolto por um material isolante na forma cilíndrica, e externamente envolto por uma trança metálica.

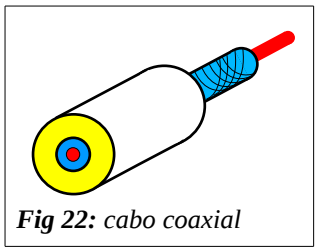


Fig 22: *cabo coaxial*

O cabo coaxial, por ser menos suscetível a interferências eletromagnéticas externas, oferece uma maior banda passante em relação ao par trançado, porém tem a desvantagem de ser mais caro.

A *fibra ótica* é o meio pelo qual os sinais binários são conduzidos sob a forma de impulsos luminosos. A luz visível é uma onda eletromagnética cuja frequência está entre 10^{14} e 10^{15} Hz, e por isso neste suporte de transmissão a banda passante potencial é bastante grande, ficando na faixa dos Gbps (10^9 bits/s).

O suporte de transmissão à base de fibra ótica é composto por três elementos: o próprio suporte de transmissão (a fibra), o dispositivo de emissão e o dispositivo de recepção.

A fibra ótica é constituída de um cilindro de fibra de vidro bem fino envolvido por uma capa, o dispositivo emissor consiste de um LED (*Light Emitting Diode*) ou diodo laser e o dispositivo de recepção é constituído por um fotodiodo ou de um fototransistor. A vantagem da fibra é oferecer alta taxa de transmissão em redes de comunicação em longa distância.

As fibras óticas precisam também de alimentação elétrica para amplificar o sinal ótico que reduz com a distância percorrida. No caso de fibras intercontinentais, o cabo submarino é alimentado com energia elétrica de alta voltagem para possibilitar esta amplificação em longas distâncias.

7.5.1.2 - Suporte de transmissão com ausência guia físico

Com exemplo de suporte de transmissão com ausência de guia físico temos o *wireless*.

As redes *wireless* usam frequências de rádio na faixa entre KHz (10^3) e GHz (10^9) ou mesmo infravermelho (THz, 10^{12}).

Pela própria natureza, as redes *wireless* são adequadas tanto para ligações ponto a ponto quanto ligações multiponto.

A principal vantagem do uso de redes *wireless* é dispensar a necessidade de cabeamento, porém como o meio de transmissão é compartilhado por várias estações, faz-se necessário um método para disciplinar esse comportamento.

Alguns desses métodos são FDM (*Frequency Division Multiplexation*), TDM (*Time Division Multiplexation*), SDM (*Space Division Multiplexation*) e CDM (*Code Division Multiplexing*). O GSM (*General System for Mobile Communications*) usa FDM e TDM combinados.

O método SDM pode ser usado de duas maneiras: a primeira usa antenas direcionais e sinais de rádio de alta frequência concentrados em feixe e a segunda estrutura a rede em células. Neste segundo caso, por não ser um feixe direcionado, a intensidade do sinal cai rapidamente à medida que uma estação se afasta da transmissora.

Uma consideração muito importante no *wireless* é a segurança, pois o sinal de rádio pode ser detectado por receptores não autorizados.

7.5.1.3 - Aspectos da transmissão de dados

Um canal de transmissão de dados a nível de *bit* envolve transmissor, receptor, destinatário e suporte de transmissão. No meio desse processo está sempre presente o ruído.

Fig 23: *transmissão ponto a ponto unidirecional*

A transmissão pode ser unidirecional e bidirecional. Se a transmissão ocorrer somente numa direção o canal é chamado de *simplex*, se o mesmo canal transmite e recebe mas não simultaneamente, o canal é *half-duplex*, se transmite e recebe simultaneamente, o canal é *full-duplex*.

full-duplex		dois canais, transmissão e recepção simultaneamente
half-duplex		um canal para transmissão e recepção, porém não simultaneamente
simplex		transmissão ou recepção somente numa direção

Tabela 3: *canais de transmissão e recepção*

Para finalizar o item *Camada física*, convém lembrar que o meio físico envolve também os conectores e *path panel*.

Como exemplo de equipamento de camada 1 temos o *hub*, que também é um concentrador de rede.

7.5.2 - Camada de enlace de dados

A camada de enlace tem a função de oferecer uma forma de comunicação confiável entre entidades da camada de rede. As funções da camada de enlace de dados são *agrupar bits em quadros*, *detecção e correção de erros de transmissão*, *controle de fluxo* e *controle de acesso ao meio*.

7.5.2.1 - Conceito de quadro

Um quadro consiste de uma sequência demarcada de bits (com início, meio e fim), e que contém centenas de *bytes*. Tipicamente um quadro tem 1500 bytes, pois este é o tamanho padrão (MTU[4]) do quadro ethernet.

Com o objetivo de permitir um controle de erros mais eficiente, aos quadros são adicionados códigos especiais de controle de erro. Dessa forma, o receptor poderá verificar se o código enviado no contexto de um quadro contém erros ou não. Num caso mais simples, a delimitação do quadro pode ser feita a partir da contagem de caracteres, onde em cada quadro vai um conjunto de caracteres especiais que indicam o número de caracteres nesse quadro.

Porém, mesmo assim devido a algum erro o próprio caractere que define a delimitação poderia ser modificado.

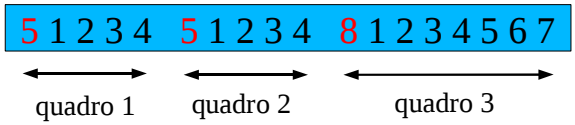


Fig 24: *controle de erro nos quadros*

A camada de enlace oferece serviços para a camada de rede acima dela, e esses serviços podem ser classificados em três categorias: *sem conexão e sem reconhecimento*, *sem conexão e com reconhecimento* e *orientado à conexão*.

No *serviço sem conexão e sem reconhecimento*, a máquina fonte simplesmente envia um quadro à máquina destinatária.

No *serviço sem conexão e com reconhecimento*, o nó processador destinatário irá enviar um quadro de reconhecimento à fonte, notificando o recebimento do quadro. Esse procedimento permite que o nó fonte retransmita o quadro caso não receba a notificação (reconhecimento) do nó destino, isso após algum tempo de espera.

Já o *serviço com conexão* é mais sofisticado, pois define a necessidade do estabelecimento da conexão previamente. Nesse caso, cada quadro é numerado e, no recebimento, os quadros serão ordenados da mesma maneira conforme enviados. Quadros danificados serão

4 MTU: *Maximum Transmission Unit*, unidade máxima de transmissão.

reenviados.

O serviço com conexão tem três etapas:

- *estabelecimento da conexão*, onde são definidos parâmetros relativos a essa conexão;
- *transmissão de dados*, onde ocorre a transferência dos dados;
- *liberação da conexão*, marcando o final do diálogo.

7.5.2.2 - Detecção e correção de erros

Por mais confiável que seja o suporte de transmissão, eventualmente ocorrem erros na transmissão dos sinais, daí a necessidade de implementar um controle de erros.

Para a correção de erros, existem duas técnicas. A primeira, pouco utilizada, consiste na introdução de informações redundantes nos quadros para permitir ao receptor reconstituir dados danificados a partir unicamente dessa informação recebida. A segunda técnica consiste em adicionar aos quadros um conjunto de informações que permita identificar a ocorrência de erro e então requisitar a retransmissão do quadro. Ao primeiro caso é dado o nome de *código corretor*, e no segundo o nome é *código detector*.

Para o caso do código de detecção de erro, uma técnica simples é a utilização de um *bit* de paridade, outro método é o CRC (*Cyclic Redundancy Code*).

7.5.2.3 - Controle de fluxo

O controle de fluxo contorna o problema gerado no caso do transmissor enviar quadros mais rapidamente que o receptor possa aceitar.

A maioria dos esquemas de controle de fluxo implica em regras que determinam quando o transmissor poderá enviar o quadro seguinte. Basicamente, implica num número máximo de quadros que o transmissor pode enviar sem receber reconhecimento do recebimento por parte do receptor.

7.5.2.4 - Controle de acesso ao meio

A camada de enlace convencionalmente está dividida em duas subcamadas do IEEE 802: *LLC* e *MAC*.

A subcamada LLC (*Logical Link Control*) estabelece e mantém *links* (enlaces) entre dispositivos em comunicação.

A subcamada MAC (*Media Access Control*) controla os meios pelos quais vários dispositivos compartilham o mesmo canal de transmissão.

Por último, a nível de *software* a camada de enlace usualmente aparece como um driver de dispositivo de Sistema Operacional.

Como exemplo de equipamento de camada 2 temos o *switch*, que também é um concentrador de rede.

7.5.3 - Camada de rede

Essa camada assegura o transporte de pacotes do sistema fonte ao destinatário, usando uma trajetória apropriada.

A camada de rede é a mais baixa que lida com a transmissão fim a fim. As suas duas funções essenciais são *roteamento* e *controle de congestionamento*.

7.5.3.1 - Organização interna da camada de rede

Existe uma analogia entre os circuitos físicos estabelecidos pelo sistema telefônico e o circuito virtual da conexão. Da mesma forma, existe uma analogia entre telegramas e datagramas.

Quando se estabelece uma conexão, é escolhida uma rota entre o nó processador de origem e o de destino, e essa rota é utilizada pelo tráfego que flui nessa conexão. Ou seja, no estabelecimento da conexão foi definido um circuito virtual para os pacotes de dados.

No caso do datagrama, nenhuma rota é previamente definida, e cada pacote enviado é roteado independentemente dos antecessores. Nesse caso existe uma melhor adaptação a falhas e congestionamentos da rede, embora o trabalho associado ao envio de dados seja maior.

Para um melhor entendimento entre circuito virtual e datagrama é apresentada a tabela abaixo.

	datagrama	***circuito virtual***
circuito	desnecessário	obrigatório
endereçamento	endereço de origem e destino obrigatório em cada pacote	basta o número do circuito virtual no pacote
roteamento	pacote roteado independentemente	rota escolhida no estabelecimento do circuito virtual
falhas no roteador	só perde os pacotes durante a falha	encerra todos os circuitos virtuais
controle de congestionamento	difícil	fácil se forem alocados *buffers* para cada circuito virtual

Tabela 4: *comparação entre datagrama e circuito virtual*

Um ponto importante a considerar é que já estamos tratando de WAN onde temos o conceito de *sub net* ou sub-rede, que consiste de dois componentes distintos:

- *Linhas de transmissão* (transmissão de *bits*);
- *Elementos de comutação* (chaveamento ou roteamento).

Em geral uma sub-rede, que na maioria dos casos é uma LAN, está ligada num roteador. Mas há casos em que um *host* se liga diretamente a um roteador.

A camada de rede controla a operação da sub-rede, e a sua principal tarefa é controlar como os pacotes de informação são roteados da fonte para o destino.

7.5.3.2 - O endereçamento de rede

Basicamente, dois tipos de endereçamento são possíveis: *hierárquico* e *horizontal*.

No *endereçamento hierárquico* o endereço é constituído de acordo com os endereços correspondentes aos vários níveis de hierarquia a que faz parte. Como exemplo temos o protocolo IP da ARPANET, onde está definido o número da rede, o número do módulo processador dentro dessa rede e o número da porta.

Por exemplo, 192.168.74.208:80 onde 192.168.74 define a rede, 208 o endereço específico desse *host* nessa rede e 80 o número da porta TCP.

No *endereçamento horizontal* os endereços não tem relação alguma com o lugar onde estão as entidades dentro dessa rede. Este esquema é usado em LANs (ex: IEEE802) e tem como vantagem facilitar a reconfiguração da rede sem necessitar alterar os endereços das estações.

7.5.3.3 - Função de roteamento

A camada de rede determina a série de saltos que os pacotes darão ao longo de sua trajetória pelo inter rede, e esta decisão pode ou não levar em consideração a situação da rede, do ponto de vista do tráfego.

Existem duas classes principais para os algoritmos de roteamento, os *adaptativos* e os *não adaptativos* (ou de rota fixa).

Os algoritmos *não adaptativos* fazem o roteamento estático, pois não levam em consideração a situação do tráfego na rede.

Os algoritmos *adaptativos* operam no conceito de roteamento dinâmico, pois levam em consideração modificações da topologia e do tráfego real, além de decidir o melhor caminho para os pacotes de dados. No caso do encaminhamento adaptativo, é necessário manter tabelas atualizadas com informações sobre a carga na rede. Como exemplo, essas tabelas poderiam ser mantidas atualizadas dinamicamente pelo RIP (*Routing Information Protocol*).

O conceito de "caminho mais curto" envolve o número de saltos, distância geográfica e retardo na transferência do pacote.

7.5.3.4 - Controle de congestionamento

O processo de congestionamento consiste basicamente de uma realimentação positiva, onde o número de mensagens tende a crescer se a rede está congestionada. Esse processo é

semelhante a um congestionamento de trânsito.

Uma função de controle de congestionamento é a *pré alocação de buffers*, que consiste na alocação de um determinado número de *buffers* em cada nó na rede no estabelecimento da conexão, e que serve para armazenamento dos pacotes a serem transmitidos pelo circuito virtual.

Outra função de controle é a *destruição de pacotes*, onde na ausência do *buffer* o pacote é então destruído pois não poderá ser armazenado.

No entanto, essa destruição deve seguir uma certa disciplina, por exemplo destruir um pacote de reconhecimento não é uma boa solução pois esse pacote poderia permitir a liberação de um *buffer*.

7.5.3.5 - Ligações inter rede

Quando se trata de comunicação entre nós em redes diferentes (LANs, MANs, WANs), diversos protocolos - e problemas - estão envolvidos nessa comunicação inter redes (inter net ou internet).

Devido às diferentes tecnologias para essas diferentes redes que se interligam, é necessário o uso de equipamentos que façam as conversões necessárias à medida que esses pacotes são transferidos de uma rede para outra. Alguns nomes para esses equipamentos são:

- Repetidor, opera na *camada física*, copia *bits* entre segmentos de cabo (*hub* também um equipamento que opera na camada 1);
- Ponte (*bridge*), opera na *camada de enlace*, armazena e remete quadros entre LANs. O *switch* é um equipamento que normalmente também opera na camada 2, mas se tiver módulo de roteamento pode operar na camada 3 (que é necessário se for o caso de criar VLANs);
- Roteador (*router*), opera na camada de rede, encaminha pacotes entre diferentes redes (*router* é um equipamento que opera na camada 3).

7.5.4 - Camada de transporte

A camada de transporte tem por função transferir informações do sistema emissor para o sistema receptor. Essa transferência é independente da natureza da informação ou rede que suporta essa comunicação. Cabe à camada de transporte estabelecer e encerrar conexões além de controlar o fluxo fim a fim.

7.5.4.1 – Serviços oferecidos pela camada de transporte

Similar à camada de rede, a camada de transporte pode fornecer serviços sem conexão e serviços orientados à conexão. Desse modo, a necessidade da camada de transporte se justifica pela necessidade de serviços de supervisão da camada de rede, só que agora do ponto de vista das entidades efetivamente envolvidas nessa comunicação. Afinal, a camada de transporte é a mais baixa que possibilita comunicação fim a fim (através de *headers*), pois até a camada de rede as comunicações ocorrem ponto a ponto.

A camada de transporte efetua um "isolamento" entre os níveis de 1 a 3 e 5 a 7, onde os 4 primeiros são mais orientados ao transporte da informação e os 3 níveis superiores mais orientados às aplicações.

Podemos sintetizar o serviço fornecido pela camada de transporte como um supervisor da qualidade do serviço da camada de rede: se a rede for confiável, pouco restará a ser feito pela camada de transporte.

O conceito de qualidade de serviço QoS (*Quality of Service*) é um aspecto importante nessa camada, entre os quais destacam-se os seguintes parâmetros:

- retardo no estabelecimento de uma conexão: tempo decorrido entre solicitação e o recebimento da confirmação;
- probabilidade de falha no estabelecimento da conexão: a conexão não se estabelece dentro de um período pré estabelecido;
- throughput: é o fluxo de dados para cada direção;
- retardo de trânsito: é o tempo decorrido desde o envio da mensagem até o recebimento pelo nó destino;
- taxa de erros residuais: é o percentual de mensagens perdidas ou corrompidas de um total enviado;
- proteção: protege os dados contra leitura ou escrita por parte de terceiros;
- prioridade: determina quais conexões são mais importantes;
- resiliência: é probabilidade de finalizar uma conexão devido a problemas ou congestionamento.

7.5.4.2 – Negociação de opção

Negociação de opção é a determinação dos valores mínimos para os parâmetros QoS no estabelecimento da conexão. Uma vez negociado será mantido durante toda a duração da conexão.

7.5.4.3 – Multiplexação e splitting

Quando o canal de comunicação (banda passante) for maior que o necessário para a conexão de transporte, podemos subdividir esse canal em vários outros menores, um para cada conexão de transporte.

Por outro lado, pode ocorrer o inverso, que é a banda passante menor que o necessário para a conexão de transporte, nesse caso pode ser usada a divisão (splitting**)** da conexão de transporte, mas isso somente no caso do nó na rede possuir mais de um canal de saída no nível físico.

7.5.5 - Camada de sessão

A camada de sessão é a responsável pelo estabelecimento de sessões que permitem o transporte ordinário de dados (assim como a camada de transporte), porém com alguns serviços mais refinados, que podem ser úteis em algumas aplicações.

Alguns serviços que a camada de sessão deve prover são *gerência do controle do diálogo*, *sincronização* e *gerenciamento de atividades da camada de sessão*.

7.5.5.1 – Gerência do controle de diálogo

A troca de informações entre entidades em um circuito *half-duplex* deve ser controlada através da utilização *data tokens* (ficha de dados). A camada de sessão é responsável pela posse e entrega desses *data tokens*, ajudando a controlar de quem é a vez de transmitir. Isso é negociado no início da sessão e não é usado no circuito *full-duplex*.

7.5.5.2 – Sincronização

Utiliza-se o conceito de ponto de sincronização para evitar, por exemplo, a perda de um volume de dados muito grande que está sendo transmitido numa rede não confiável. O ponto de sincronização corresponde a marcas lógicas posicionadas ao longo do diálogo. Toda vez que um processo cliente recebe um ponto de sincronização ele deve enviar uma resposta, confirmando que os dados naquele segmento foram recebidos. Caso a transmissão seja interrompida, ela poderá ser reiniciada a partir do último ponto de sincronização confirmado.

O mecanismo de sincronização define dois tipos distintos de pontos de sincronização: pontos de *máximos* e *mínimos*. Os pontos de máximo delimitam trechos chamados diálogos (por exemplo, o capítulo de um livro) e os pontos de mínimo são utilizados para separar porções menores de informação (num livro, seriam as páginas).

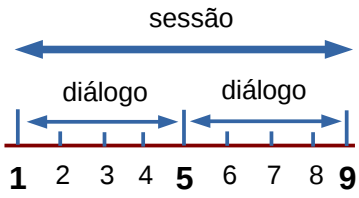


Fig 25: *pontos de sincronização 1, 5 e 9*

A capacidade de ressincronização está associada ao ponto de sincronização máximo, pois a partir daí é impossível recuperar a informação. O ponto máximo é visto como uma *fronteira de proteção da informação*.

7.5.5.3 – Gerenciamento de atividades da camada de sessão

O controle de atividades está baseado no conceito de decomposição do fluxo de dados em atividades independentes umas das outras. Por exemplo, quando temos a transferência simultânea de vários arquivos, cada arquivo deve ser separado dos demais e tratado como uma atividade.

7.5.6 - Camada de apresentação

A camada de apresentação, ao contrário das camadas inferiores, já não se preocupa com os dados a nível de *bits*, mas sim com a sua sintaxe e a sua representação. Nela é definida a sintaxe abstrata, que é a forma como os tipos e os valores dos dados são definidos, independentemente do sistema computacional utilizado e da sintaxe de transferência, que é a maneira como é realizada esta codificação. Por exemplo, através da sintaxe abstrata define-se que

um caractere deve ser transmitido, então a sintaxe de transferência especifica como este dado será codificado em ASCII ou EBCDIC ao ser entregue à camada de sessão.

ASCII (*American Standard Code for Information Interchange*) é conjunto de normas de codificação de caracteres mediante caracteres numéricos e EBCDIC (*Extended Binary Decimal Interchange Code*) é conjunto de normas de codificação binária de caracteres mediante números.

A tarefa da camada de apresentação está relacionada à representação dos dados a serem transmitidos, tendo a função de *conversão de dados*, *compressão de dados* e *criptografia*.

7.5.6.1 – Compressão de dados

As técnicas de compressão de dados estão baseadas em três diferentes aspectos da representação de dados, que é a *limitação do alfabeto*, *frequência relativa dos símbolos* e *contexto de aparecimento dos símbolos*.

Limitação do alfabeto: a ideia da limitação do alfabeto é efetivamente utilizar todas as possibilidades de construção de uma palavra. Por exemplo, alocar 5 *bytes* para uma palavra em português é um desperdício, pois nem todas as possibilidades serão exploradas pela linguagem, daí o desperdício. Por exemplo, A B C D E e C A R R O.

Frequência relativa dos símbolos: essa técnica consiste em codificar os símbolos mais frequentes por códigos curtos. Codifica os dados de tal forma que os símbolos ou sequências de símbolos mais frequentes sejam representados de forma especial, simplificando a quantidade de informação. Por exemplo, no inglês a palavra "E" é 100 vezes mais frequente que "Q", "THE" é 10 vezes mais frequente que "BE".

Contexto de aparecimento dos símbolos: a codificação baseada em contexto dos símbolos pode ser implementada de várias maneiras, a maioria dependente da própria informação a ser transmitida. Por exemplo, dada a sequência binária "000100000100001", nela as sequências de zeros poderiam ser resumidas a "011 101 100", onde 011 é 3 em decimal, 101 é 5 e 100 é 4.

7.5.6.2 – Criptografia

O processo de criptografia consiste em codificar as mensagens através de uma função parametrizada por uma chave, que gera o texto criptografado ou criptograma. Quem tiver a posse da chave, poderá decodificar a mensagem. Por exemplo, dada a mensagem "CARRO", se a função parametrizada for "trocar A por B", então o criptograma será "CBRRO".

Em muitos casos usam-se duas chaves, uma pública e outra privada, que é chamado de criptografia assimétrica. A chave pública é livremente distribuída e serve para gerar o criptograma que somente poderá ser aberto por quem possui a chave privada (é o caso usado nos protocolos HTTPS e SSH).

7.5.7 - Camada de aplicação

A camada de aplicação é a que mantém o contato direto com os usuários da arquitetura

de comunicação, abrindo caminho para todos os serviços oferecidos pelas camadas inferiores.

Basicamente, as funções dessa camada são aquelas necessárias à adaptação dos processos de aplicação ao ambiente de comunicação.

7.6 - As camadas do TCP/IP

A arquitetura do TCP/IP se baseia num modelo de 4 camadas (*acesso a rede*, *internet*, *transporte* e *aplicação*), onde cada camada executa um conjunto bem definido de funções de comunicação. Ao contrário do modelo RM-OSI, não existe uma estruturação formal para cada camada. A arquitetura TCP/IP procura definir um protocolo próprio para cada camada.

7.6.1 - Camada de acesso a rede

A camada de acesso a rede é responsável pela transmissão de dados por meio de uma facilidade física chamada *meio físico*. Exemplos de conceitos tratados nessa camada incluem definições para as tecnologias de redes locais como *ethernet* e *token-ring* ou especificações para os *drivers* de sistema operacional e suas correspondentes placas de interface de rede.

A arquitetura internet TCP/IP não faz nenhuma restrição às redes que são interligadas para formar a inter rede (internet). Portanto, qualquer tipo de rede pode ser ligada, bastando para isso que seja desenvolvida uma interface que compatibilize a tecnologia específica da rede com um protocolo IP. Essa compatibilização é a principal função do nível de interface de rede, que recebe os datagramas IP do nível inter rede e os transmite através de uma rede específica. Nesse nível, para realizar essa tarefa, os endereços IPs (endereços lógicos) são traduzidos para os endereços físicos dos *hosts* ou *gateways* conectados à rede.

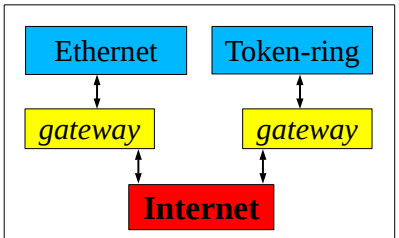


Fig 26: *gateway interconectando tecnologias ethernet e token-ring*

Para que todas estas tecnologias possam ser "vistas" pela rede internet, existe a necessidade de uma conversão de endereçamentos do formato utilizado pela sub-rede e o formato IP. Esta conversão é realizada pelos *gateways*, que tornam a interconexão das redes transparentes para o usuário. Além das conversões de protocolos, os *gateways* são responsáveis pela função de roteamento das informações entre as sub-redes.

7.6.2 – Camada internet

A camada internet, também chamada inter rede, é equivalente à camada de rede do modelo OSI. A sua responsabilidade é transferir dados através da inter rede, desde o nó de origem até o nó destino. Essa camada recebe pedidos da camada de transporte para transmitir datagramas que, ao solicitar transmissão, informa o endereço da máquina onde o pacote deverá ser entregue. Nesta camada são especificados vários protocolos, dentre os quais se destaca o IP (*Internet Protocol*).

O IP é um protocolo cuja função é transferir blocos de dados denominados datagramas da origem até o destino, podendo passar inclusive por várias sub-redes (a origem e o destino são *hosts* identificados por endereços IPs). Na mesma rede, duas máquinas (ou nós) nunca podem ter o mesmo endereço IP.

Para garantir que os *gateways* encaminhem as mensagens corretamente, é utilizado um controle de verificação de cabeçalhos (*headers*).

7.6.2.1 – Endereços IPs e classes

Os IPs são números que representam os endereços. Na versão 4 (ipv4, a mais usada atualmente) são gastos 32 *bits* (4 *bytes*) para representar os endereços IPs. Normalmente são usados nessa representação quatro octetos separados por pontos, por exemplo 192.168.1.10. A primeira parte do endereço IP representa uma rede específica no inter rede, e a segunda parte identifica um *host* dentro dessa rede.

Por exemplo, a representação do endereço IP decimal 192.168.1.10 na base binária é 11000000.10101000.00000001.00001010.

```
                                        network ID
                                   ------------------------------------
192  .  168  .  1  .  10  =  11000000.10101000.00000001.00001010
------  -------  ------  ------                                   ------------
8 bits  8 bits  8 bits  8 bits => 32 bits = 4 bytes                    host ID
```

O primeiro octeto do exemplo acima tem valor decimal 192 (acima), que é obtido da conversão binária:

$$1x2^7 + 1x2^6 + 0x2^5 + 0x2^4 + 0x2^3 + 0x2^2 + 0x2^1 + 0x2^0 = 128+64+0+0+0+0+0+0 = 192$$

O processo inverso, isto é, dado um número na base decimal, para obter o valor na base binária usa-se a divisão contínua por dois, conforme esquema abaixo. Neste esquema, o resto da divisão é que determina o valor na base binária.

Em notação decimal, 8 *bits* possibilita um intervalo compreendido entre 0 e 255, pois $2^8 = 256$. Desse modo, o maior endereço IP é "255.255.255.255" e o menor é "0.0.0.0".

Cada nó (*host* ou computador) numa mesma rede deve ter um endereço IP único, e no conjunto esses IPs devem estar ajustados para as diversas configurações possíveis dessa rede.

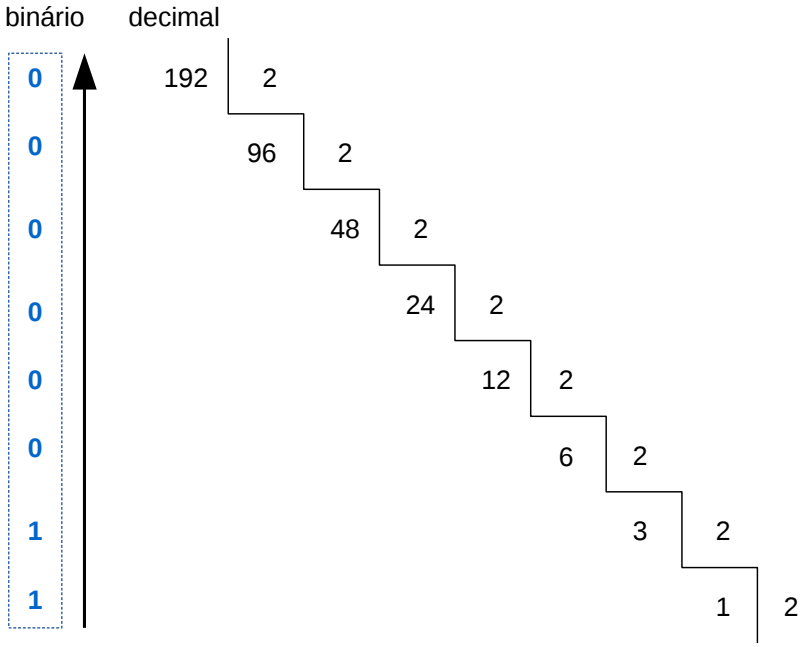


Fig 27: *conversão de decimal para binário*

As classes de endereçamento IP foram originalmente divididas em A, B, C, D e E, e a tabela 4 abaixo mostra que a separação dessas classes está definida pelos quatro primeiros *bits* da esquerda do endereço IP.

Classe	bits da esquerda (*leftmost bits*)	Início dos endereços	Final dos endereços
A	0xxx	0.0.0.0	127.255.255.255
B	10xx	128.0.0.0	191.255.255.255
C	110x	192.0.0.0	223.255.255.255
D	1110	224.0.0.0	239.255.255.255
E	1111	240.0.0.0	255.255.255.255

Tabela 5: *classes de endereçamento IP*

Por exemplo, o endereço IP 192.168.1.10 é da classe C, 10.10.15.30 é classe A. Esses dois endereços são exemplos de IPs reservados ou não válidos, pois somente têm validade dentro de uma LAN, não sendo possível roteá-los através do inter redes.

Alguns endereços IPs possuem significado especial:

- 0: significa a própria rede ou sistema. Por exemplo, o endereço 0.0.0.13 referencia a estação 13 da rede local. A faixa de endereços que vai de 0.0.0.0 até 0.255.255.255 não serve para nenhuma função particular no IP e por isso não pode ser considerada parte da classe A;

- 127.0.0.0: referencia a estação em análise;
- 127.0.0.1: é conhecido com *loopback* e utilizado em processos de diagnóstico, por exemplo para testar a interface de rede. A faixa de IPs entre 127.0.0.0 e 127.255.255.255 está reservada para propósitos de *loopback*. Como esses endereços não podem ser usado externamente ao nó (isto é, na rede), os endereços *loopback* não podem ser considerados parte da classe A.

Os endereços classe A são usados para redes muito grandes, normalmente ligadas a funções educacionais e científicas.

Os endereços classe B também são usados em redes grandes, e historicamente foram atribuídos a instituições que possuíam um perfil disseminador de tecnologia e assim pudessem de alguma forma distribuir suas redes entre instituições e empresas contribuindo para o desenvolvimento de uma grande rede mundial.

Os endereços classe C são os mais difundidos pois permitem redes com 256 IPs, que aparenta ser um número conveniente para gerenciamento e implantação de sistemas de informação.

Os endereços classe D são reservados para *multicast* utilizados, entre outras, nas aplicações de videoconferência e multimídia. *Multicast* é um mecanismo para definir grupos de nós e enviar mensagens IP para esses grupos ao invés de nós na própria LAN (que é *broadcast*). O *multicast* também não envia mensagens para um único nó (que é *unicast*). A classe D é usada principalmente em redes de pesquisa e desenvolvimento. Da mesma forma que a classe E, endereços da classe D não podem ser usados em nós ordinários na inter redes.

Os endereços classe E são reservados e usados apenas em experimentação e desenvolvimento.

Os padrões do IP também definem faixas de IPs reservados para redes privadas (intranets), também chamados de endereços IP privados. Essas faixas de IPs são reservadas nas classes A, B e C.

Classe	***Rede* (*máscara*)**	***Início da faixa reservada***	***Final da faixa reservada***
A	10.0.0.0/8 (255.0.0.0)	10.0.0.0	10.255.255.255
B	172.16.0.0/12 (255.240.0.0)	172.16.0.0	172.31.255.255
C	192.168.0.0/16 (255.255.0.0)	192.168.0.0	192.168.255.255

Tabela 6: *faixas de IPs reservados nas classes de endereçamento*

Na rede, os nós são efetivamente livres para usar endereços da faixa reservada, porém esses IPs não são roteados através do inter redes. Se isto for necessário usando-se IPs da faixa reservada, deve-se usar também técnicas como NAT (*Network Address Translation*), *firewall* ou *proxy*.

Como o endereço IP é constituído por quatro octetos, é preciso converter esse número de decimal para binário a fim de facilitar a visualização do endereço. Por exemplo:

0	00000000
255	11111111
128	10000000
127	01111111
252	11111100
253	11111101
254	11111110
210	11010010

No caso, 11010010 corresponde ao decimal 210 pois para traduzir o endereço binário para decimal foi efetuada a soma:

$1x2^7 + 1x2^6 + 0x2^5 + 1x2^4 + 0x2^3 + 0x2^2 + 1x2^1 + 0x2^0 =>$
$128 + 64 + 0 + 16 + 0 + 0 + 2 + 0 = 210$

Além das classes de endereços, os IPs também distinguem o endereço da rede (identificador de rede - *network* ID) e o endereço da máquina local (identificador de máquina local - *host* ID).

Classe	***network ID***	***host ID***	***Número de redes***	***hosts em cada rede***
A	*bits* 1 a 7	*bits* 8 a 31	128 (2^7)	16772216 (2^{24})
B	*bits* 2 a 15	*bits* 16 a 31	16384 (2^{14})	65536 (2^{16})
C	*bits* 3 a 23	*bits* 24 a 31	2097152 (2^{21})	256 (2^8)

Tabela 7: *ID rede e ID máquina nas classes de endereçamento*

Além disso, o ipv4 reserva todos os *bits* 0 ou todos os *bits* 1 nos octetos para endereços especiais. Portanto, os números de máquinas/redes acima precisam ser subtraídos de 2 para obter o número de redes e máquinas. O octal que somente contiver zeros (00000000 => decimal 0) é conhecido como identificador de rede, e o que somente contiver 1 (11111111 => decimal 255) é conhecido como *broadcast*. Por exemplo, 192.168.1.0 é rede, 192.168.1.255 é *broadcast*. Para definir essa rede, podem ser usadas as duas notações abaixo:

192.168.1.0/24 ou
192.168.1.0/255.255.255.0 onde 255.255.255.0 é o *netmask* (máscara da rede).

A máscara de rede separa quais porções são network ID e quais são host ID.

Mas também existe a sublocação, que consiste em subdividir as redes em sub redes, nesse caso o identificador de rede é determinado dos *bits* que foram usados para constituir essa sub rede, e o *broadcast* é determinado dos *bits* de *hosts* ID que sobraram. Neste caso, também vale a definição acima de 0 para identificação de rede e 1 para *broadcast*.

7.6.2.2 – VLSM: sub redes

Antes da introdução do CIDR, *Classless Inter Domain Routing*, em 1993, a única maneira de segmentar as redes era pela máscara padrão dada pela classe de IP:

Classe A: 0 – 127 máscara: 255.0.0.0 (/8)
Classe B: 128 – 191 máscara: 255.255.0.0 (/16)
Classe C: 192 – 223 máscara: 255.255.255.0 (/24)

Isto era chamado de *classfull network* e, obviamente, limitava em muito as possibilidades de segmentação da rede em sub redes, além de dificultar a tarefa de roteamento pois cada rede precisava da sua própria rota.

Com CIDR, ao invés de ficar limitado a apenas máscaras /8, /16 e /24 (um octeto completo), pode-se usar /9, /10, /18, /25, /26, /27, etc.

VLSM, *Variable-Length Subnet Masking* (mascaramento de subredes em tamanhos variáveis), está muito próximo do CIDR. Tão próximo que até causa confusão. Basicamente, VLSM é para a segmentação da rede, enquanto CIDR é para agregação de prefixos de roteamento.

Por segmentação de rede entende-se, por exemplo, dividir uma rede /24 em duas /25. Por agregação de prefixos, por exemplo, a junção de duas redes /24 numa /23.

Um exemplo de sub rede é mostrado abaixo:

10.1.2.0/8 (10.1.2.0/255.0.0.0) => 10.1.2.0/24 (10.1.2.0/255.255.255.0)

Repare que neste exemplo a classe de IP é A, mas devido ao uso da máscara em classe C, funcionalmente esta rede é uma classe C. Neste caso, a sub rede foi construída alterando a máscara original de classe A (8 bits) em dois octetos (um octeto é um espaço de 8 bits no endereço IP).

Um caso mais avançado implica em tomar um a um os bits de *host* ID e transferi-los para o *network* ID, deste modo segmentando a rede.

A criação de sub redes a partir de uma rede primária é um procedimento típico na área de redes. O objetivo desta segmentação é permitir uma melhor performance da rede em termos organizacionais, estruturais e funcionais.

A ideia básica é acrescentar alguns *bits* ao identificador de rede, *bits* esses tomados do identificador de *hosts*. Os endereços permitidos serão os restantes no octeto.

A máscara de sub rede é um endereço de 32 *bits* usado para bloquear (mascarar) uma parte do endereço IP para se poder distinguir a parte do identificador de rede (*network ID*) da parte do identificador de *host* (*host ID*). Depois desse "empréstimo", as possibilidades de endereços de *host* que sobrarem e que corresponderem a apenas uma sequência de zeros, serão os identificadores de rede, e os que corresponderem a uma cadeia de apenas dígitos 1 serão os *broadcasts*.

Dada a rede 192.168.1.0, se quisermos segmentá-la em 4 sub redes, precisamos tomar emprestado 2 *bits* do identificador de *hosts*, pois $2^2 = 4$. Por exemplo:

192.168.1.0/24 = **11000000.10101000.00000001**.00000000

onde os bits representados em negrito são identificadores de rede (24 *bits*).

No caso de 4 sub redes, temos:

192.168.1.0/26 = **11000000.10101000.00000001.00**000000
192.168.1.64/26 = **11000000.10101000.00000001.01**000000
192.168.1.128/26 = **11000000.10101000.00000001.10**000000
192.168.1.192/26 = **11000000.10101000.00000001.11**000000

onde estamos usando 26 *bits* para a definição da rede.

Nesse caso, os endereços de *broadcast* são, respectivamente:

192.168.1.63 = 11000000.10101000.00000001.00111111
192.168.1.127 = 11000000.10101000.00000001.01111111
192.168.1.191 = 11000000.10101000.00000001.10111111
192.168.1.255 = 11000000.10101000.00000001.11111111

Num outro exemplo, considere uma rede classe C segmentada em 8 sub redes, nesse caso é necessário tomar 3 *bits* do identificador de *hosts* pois $2^3 = 8$. Cada uma dessas sub redes terá 32 endereços (256 / 8), mas o número total de *hosts* em cada um delas será 30, pois um número IP foi gasto com o identificador de rede e outro com o *broadcast*.

sub rede	faixa de IPs	*broadcast*
192.168.1.0	0-31	192.168.1.31 (00011111)
192.168.1.32	32-63	192.168.1.63 (00111111)
192.168.1.64	64-95	192.168.1.95 (01011111)
192.168.1.96	96-127	192.168.1.127 (01111111)
192.168.1.128	128-159	192.168.1.159 (10011111)
192.168.1.160	160-191	192.168.1.191 (10111111)
192.168.1.192	192-223	192.168.1.223 (11011111)
192.168.1.224	224-255	192.168.1.255 (11111111)

A representação dessas sub redes também poderia ser feita usando o número de *bits* empregados nessas construções, por exemplo a rede 192.168.1.0/255.255.255.0 equivale a 192.168.1.0/24 pois foram empregados 24 *bits* na definição da rede. Já no caso das 8 sub redes acima a representação é 192.168.1.0/27, 192.168.1.32/27, 192.168.1.64/27, etc.

O número máximo de subdivisões de uma rede de classe C é mostrado abaixo:

Nº de sub redes	***Nº de bits tomados***	***Máscara - bits (decimal)***	***Nº de hosts***
2	1	10000000 (255.255.255.128)	126
4	2	11000000 (255.255.255.192)	62
8	3	11100000 (255.255.255.224)	30
16	4	11110000 (255.255.255.240)	14
32	5	11111000 (255.255.255.248)	6
64	6	11111100 (255.255.255.252)	2

Tabela 8: *Número máximo de subdivisões de uma rede de classe C*

O exemplo de segmentação acima também pode ser aplicado a classes A e B. A diferença será no número de *hosts* por sub rede.

Por exemplo, partindo da rede 172.16.0.0/16, de 65534 *hosts* (65536 – 2), onde 65536=256x256, segmentá-la em quatro sub redes de 16382 *hosts* (16384 -2):

172.16.0.0/16 = **10101100.00010000.**00000000.00000000

onde os bits representados em negrito são identificadores de rede (16 *bits*).

No caso de 4 sub redes, temos:

172.16.0.0/18 = **10101100.00010000.00**000000.00000000
172.16.64.0/18 = **10101100.00010000.01**000000.00000000
172.16.128.0/18 = **10101100.00010000.10**000000.00000000
172.16.192.0/18 = **10101100.00010000.11**000000.00000000

onde estamos usando 18 *bits* para a definição da rede.

Nesse caso, os endereços de *broadcast* são, respectivamente:

172.16.63.255 = 10101100.00010000.00**111111.11111111**
172.16.127.255 = 10101100.00010000.01**111111.11111111**
172.16.191.255 = 10101100.00010000.10**111111.11111111**
172.16.255.255 = 10101100.00010000.11**111111.11111111**

A tabela 9, abaixo, mostra o número de *hosts* por sub redes para segmentação em classe B.

Nº de sub redes	***Nº de bits tomados***	***Máscara - bits (decimal)***	***Nº de hosts***
2	1	255.255.128.0	32766
4	2	255.255.192.0	16382
8	3	255.255.224.0	8190
16	4	255.255.240.0	4096
32	5	255.255.248.0	2046
64	6	255.255.252.0	1022
128	7	255.255.254.0	510
256	8	255.255.255.0	254

Tabela 9: *Número máximo de subdivisões de uma rede de classe B*

7.6.2.3 – CIDR: agregação de prefixos de roteamento

Agregação de prefixos de roteamento traz como primeiro benefício a eficiência ganha nos roteadores, pela economia de memória, de processamento e redução de informações de rotas.

Neste contexto, *supernetting* é um bloco contíguo de sub redes (ou redes) roteado como uma simples rede. Isso alivia os roteadores de tabelas de roteamento excessivamente grandes.

Deste modo, para rotear duas redes classe C, por exemplo 192.168.2.0/24 e 192.168.3.0/24, não precisa incluir duas rotas. Basta considerar estas duas redes como um agregado, pelo uso da máscara /23.

192.168.2.0/24
192.168.3.0/24 } 192.168.2.0/23

192.168.2.0/24 = **11000000.10101000.00000010**.00000000
192.168.3.0/24 = **11000000.10101000.00000011**.00000000

192.168.2.0/23 = **11000000.10101000.0000001**0.00000000

onde estamos usando 23 *bits* para a definição da rede.

Neste caso, o endereço de *broadcast* é 192.168.3.255, e a faixa de *hosts* vai de 192.168.2.1 a 192.168.3.254.

Neste outro exemplo, as quatro redes abaixo também podem ser agregadas usando a máscara /22. Como $2^2 = 4$, então para agregar 4 redes é necessário tomar 2 bits do identificador de rede (*network ID*). A máscara que era de 24 bits passou para 22.

192.168.0.0/24
192.168.1.0/24
192.168.2.0/24
192.168.3.0/24 } 192.168.0.0/22

Aqui, o *broadcast* é 192.168.3.255 e a faixa de *hosts* vai de 192.168.0.1 a 192.168.3.254.

Caso seja necessário agregar redes que sejam muito diferentes entre si, ou que tenham máscaras de rede bem distintas, a solução é identificar o primeiro octeto em que apareçem estas diferenças, então converter para binário e depois analisar o que for comum.

Por exemplo, dadas as seguintes três redes abaixo, a parte comum são os dezoito primeiros bits, representados na cor **azul**:

192.168.31.0/24 = **11000000.10101000.00011111**.00000000
192.168.45.128/25 = **11000000.10101000.00101101**.10000000
192.168.27.64/27 = **11000000.10101000.00011011.010**00000

192.168.0.0/18 = **11000000.10101000.00**000000.00000000 (agregado)
192.168.63.255 = **11000000.10101000.00**111111.11111111 (broadcast)

O raciocínio acima é o mesmo empregado para a segmentação da rede, a diferença é que agora tomamos bits do *network ID* ao invés de *host ID*, portanto aumentando o número de *hosts* no agregado.

Agregação de prefixos de roteamento também é conhecido por rota sumarizada ou agregada.

Os protocolos de roteamento BGP (*Border Gateway Protocol*, que prevalece no exterior - *interdomain*) e OSPF (*Open Shortest Path First*) suportam *supernetting*, já EGP (*Exterior Gateway Protocol*) e RIP (*Routing Information Protocol*) não.

7.6.2.4 – IP versão 6 (ipv6)

Desde 1981 que é usada a versão 4 do protocolo IP (ipv4). Desde então, mesmo sofrendo correções como foi a alteração de *classfull* para *classless*, esta versão tem conseguido atender e se adequar às novas condições da internet que não tem parado de crescer. No entanto, por ser de apenas 32 bits, o número teórico máximo de endereços IP do ipv4 não atende mais à demanda, problema este que se agrava a cada dia que passa. Técnicas como NAT[3] têm sido usadas para contornar a exaustão dos endereços públicos ipv4, mas não sabemos até quando ainda será possível continuar adiando a adoção definitiva do ipv6.

Está bastante claro que o futuro exige uma nova versão, devido às limitações do ipv4.

É claro também que a nova versão não modifica radicalmente as coisas conforme vinham sendo feitas na versão anterior, apenas introduz novidades. Abaixo segue uma lista das mudanças mais importantes entre ipv4 e ipv6.

- **Aumenta o espaço de endereçamento**: de 32 para 128 bits;
- **Espaço de endereçamento hierárquico**: o tamanho do endereço foi expandido para permitir divisão hierárquica e prover um número grande de classes de endereços. Este tem sido um tema ainda bastante discutido, por remeter de volta ao *classfull*;
- **Atribuição hierárquica de endereços *unicast***: foi criado um formato global de endereço *unicast* para permitir que os endereços sejam facilmente alocados na internet inteira. Permite múltiplos níveis de redes e sub redes para ISP (*Internet Service Provider*, fornecedor de acesso à internet) e também para os demais níveis organizacionais. Também permite a geração de endereço IP baseado no endereço de hardware, tal como Ethernet MacAddress. Ao contrário do ipv4, em ipv6 não existe

3 NAT: *Network Address Translation* é técnica de tradução de endereços IP ao passar de uma rede para outra. Por exemplo, ao sair da rede interna, o pacote de dados com endereço IP privado (192.168.1.10) é traduzido para o endereço público 186.251.39.91.

suporte a *broadcast*;

- **Melhor suporte para endereçamento não-*unicast***: além de melhorar o suporte a *unicast*, foi adicionado um novo tipo de endereçamento, que é o *anycast*. Este novo tipo de endereçamento basicamente diz para direcionar a mensagem para o membro do grupo mais fácil de alcançar. Isso tem o potencial de permitir novos tipos de funcionalidades de envio de mensagens. Conceitualmente, *anycast* pode ser considerado um intermediário entre *multicast* e *unicast*. Por exemplo, *unicast* diz "envie para este endereço específico", *multicast* diz "envie para cada membro deste grupo" e *anycast* diz "envie para qualquer um que seja membro deste grupo". Deste modo, *anycast* pode ser normalmente considerado como "envie para o membro mais próximo deste grupo";
- **Conectividade fim a fim**: após ipv6 ser completamente implementado, cada sistema terá um endereço IP único e poderá cruzar a internet sem uso de NAT ou outro componente de tradução. Cada *host* poderá acessar diretamente o outro;
- **Autoconfiguração e renumeração**: foi incluída uma provisão para permitir fácil autoconfiguração de *hosts* e renumeração de endereços IP em redes e sub redes, conforme for necessário. Um recurso implementado em ipv6 permite aos dispositivos se autoconfigurarem independentemente, sem necessidade de DHCP[4];
- **Novo formato de datagrama**: foi redefinido o formato do datagrama IP e incluídas novas capacidades. O principal cabeçalho de cada datagrama IP foi simplificado e também foi incluído suporte para facilmente estender o cabeçalho dos datagramas que necessitarem de mais informações de controle;
- **Suporte para qualidade do serviço**: foi incluída a funcionalidade QoS (*Quality of Service*, qualidade do serviço) nos datagramas ipv6 para permitir melhor suporte para multimídia e outras aplicações que requerem QoS;
- **Suporte à segurança**: o suporte à segurança foi projetado para o ipv6 pelo uso de autenticação e cabeçalhos com extensão de encriptação, além de outras funcionalidades. É usado IPSec (*IP Security Protocols*) similar ao ipv4;
- **Fragmentação atualizada e procedimentos de remontagem**: a nova maneira de trabalhar a fragmentação e remontagem de datagramas mudou com o ipv6 para incrementar a eficiência do roteamento e melhor refletir a realidade das redes atuais;
- **Modernizado o suporte ao roteamento**: o protocolo ipv6 foi desenvolvido para suportar os sistemas de roteamento modernos e permitir a expansão à medida que a internet cresce;
- **Capacidades de transição**: como foi reconhecido desde o início que a mudança de ipv4 para ipv6 seria um grande passo, o suporte à transição ipv4/ipv6 foi providenciado em numerosas áreas. Isso inclui um plano de interoperação entre redes ipv4 e ipv6 e mapeamento entre endereços ipv4 e ipv6.

Algumas mudanças dignas de nota é substituição de ICMP por ICMPv6 e a adição do NDP (*Neighbor Discovery Protocol*, protocolo de descobrimento de vizinhança). O NDP realiza diversas funções que antes eram feitas pelo ARP (*Address Resolution Protocol*) do ipv4.

Entre outras funcionalidades, o NDP tem o recurso de descobrir *hosts* e roteadores e criar uma lista de roteadores locais. Se não for usado NDP, a lista de roteadores locais precisa ser configurada manualmente.

Um esquema do *main header* (cabeçalho principal) pode ser visto abaixo:

4 DHCP: *Dynamic Host Configuration Protocol* é um protocolo padrão para distribuir dinamicamente endereços IP para interfaces e serviços. Se não for usado DHCP, o endereço IP precisa ser configurado manualmente, e neste caso é dito endereço estático.

Versão (4 bits)	Classe de tráfico (8 bits)	Rótulo de fluxo (*Flow Label*) (20 bits)	
Extensão da carga (*Payload Length*) (16 bits)		Próximo cabeçalho (8 bits)	Salto limite (8 bits)
Endereço de origem (128 bits)			
Endereço de destino (128 bits)			

Tabela 10: *Main header ipv6*

Um esquema do endereço ipv6 é mostrado abaixo:

```
      network ID                host ID
--------------------------- ---------------------------
hhhh:hhhh:hhhh:hhhh:hhhh:hhhh:hhhh:hhhh
--------------------------- ---------------------------
        64 bits                   64 bits
```

Alguns exemplos de endereços na versão ipv6 são:

fe80::278a:5bff:fe81:5093
fda8:c3fd:2177:0:278a:5bff:fe81:5093

Pode ser notado que o endereço é semelhante ao ipv4, com a diferença de não fazer subnet menor do que /64.

Em ipv6, o *network* ID é atribuído administrativamente, já o *host* ID pode ser obtido de três maneiras:

1 – usando um número gerado aleatoriamente;

2 – usando DHCPv6;

3 – usando o formato EUI-64 (*Extended Unique Identifier*), que expande o endereço físico (MacAddress) de 48 para 64 bits pela inserção de fffe no meio do endereço e inverte o sétimo bit. Por exemplo, dado o endereço MAC 44:8a:5b:94:63:9a, os passos seguidos para obter o formato EUI-64 são:

```
44:8a:5b          94:63:9a          <= divide ao meio
44:8a:5b  fffe    94:63:9a          <= insere fffe
44 = 0100 0100                      <= hexadecimal convertido para binário
46 = 0100 0110                      <= sétimo bit invertido
46:8a:5b:ff:fe:94:63:9a             <= reagrupando as partes

468a:5bff:fe94:639a                 <= host ID no formato EUI-64
```

Pelo fato da estrutura do endereço ipv6 ser muito grande, são usadas duas regras para remover zeros e simplificar o endereço. Tomando como exemplo o endereço abaixo, vamos aplicar estas regras:

2001:**0000**:**0000**:**00**b3:**0000**:5bff:fe94:639a

Regra 1: descartar os zeros à frente do número: no quarto bloco (00b3), os dois zeros devem ser omitidos. Isso leva o endereço para:

2001:0000:0000:**b3**:0000:5bff:fe94:639a

Regra 2: se dois ou mais blocos consecutivos contém zeros, devem ser omitidos e substituídos por dois dois pontos. Isso leva o endereço para:

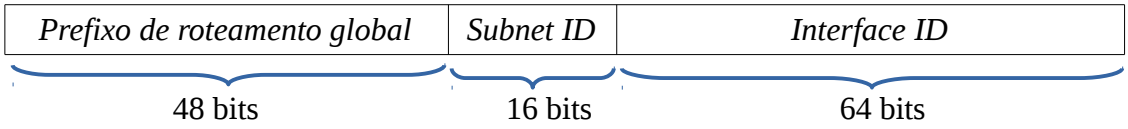
2001::**b3**:0000:5bff:fe94:639a

Depois disso, se ainda existirem blocos de zeros no endereço, ele apenas pode ser simplificado para um zero, nunca substituído por dois pontos. Isto leva o endereço para:

2001::**b3**:**0**:5bff:fe94:639a

O ipv6 define muitos tipos de endereços. Alguns exemplos são global *unicast*, *link local*, *multicast*, *anycast* e *loopback*.

Global unicast: é usado para identificar uma única interface. São endereços roteáveis na internet. Exemplo:



Os seguintes prefixos abaixo são exemplos de prefixos globais:

2400:0000::/12 APNIC (Asia-Pacific Network Information Centre)

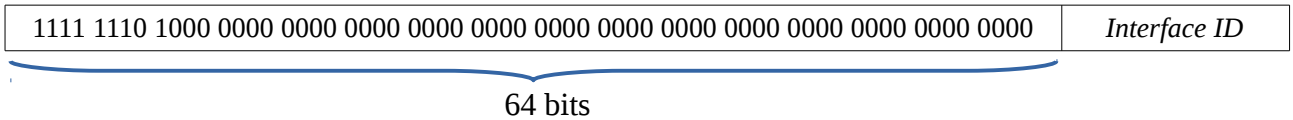
2600:0000::/12 ARIN (American Registry for Internet Numbers)

2800:0000::/12 LACNIC (Latin America and the Caribbean)

2A00:0000::/12 RIPE NCC (Regional Internet Registry for Europe, West Asia, and the former USSR)

2C00:0000::/12 AfriNIC (African Network Information Center)

Link local: são usados para permitir comunicação entre dispositivos na rede local. Estes endereços sempre iniciam por **fe80** (1111 1110 1000 0000), e os 48 bits seguintes são zero. Exemplo:



Multicast: a transmissão *multicast* envia datagramas para todas as interfaces que são parte do grupo *multicast*. O grupo é representado pelo endereço de destino do datagrama. Os endereços *multicast* iniciam por **ff**. Exemplos:

ff02::1 => todos os nós no segmento local da rede

ff02::2 => todos os roteadores no segmento local da rede

Anycast: o endereço *anycast* identifica múltiplas interfaces. Uma transmissão *anycast* envia datagramas para somente uma das interfaces associada ao endereço, e não para todas as interfaces. Esta interface é tipicamente a mais próxima, conforme definido pelo protocolo de roteamento.

Loopback: é usado por um nó para enviar o pacote para ele mesmo. Funciona do mesmo modo que no ipv4. Exemplo:

0000:0000:0000:0000:0000:0000:0000:0001/128, que é representado como **::1/128** ou simplesmente **::1**.

Em ipv6, o endereço especial que representa o *default gateway* (rota padrão) é:

::/0

Outro aspecto importante a ser notado é o formato usado com aplicações. Exempolo:

http://[2001:db8:f0b:1af0::1]/index.html <= endereço em URL

http://[2001:db8:f0b:1af0::1]:443/index.html <= endereço e porta em URL

[2001:db8:f0b:1af0::1]:21 <= porta TCP

2001:db8:f0b:1af0::1/64 <= endereço na notação CIDR

2001:db8:f0b:1af0::1%eth0 <= identificação de zona

ssh user@2001:db8:f0b:1af0::1%eth0 <= acesso SSH na zona

scp arquivo 2001:db8:f0b:1af0::1%eth0/diretorio <= transferência scp

Mas nas redes internas os endereços privados ipv4 estão sendo usados com sucesso e sem previsão de exaustão mesmo para redes muito grandes. Desse modo, parece difícil que alguém queira alterar os endereços internos de ipv4 para ipv6, especialmente porque pode muito bem continuar sendo usado NAT[5] com ipv4 nas redes internas, e migrado para ipv6 para endereços públicos.

Um ponto importante a ser considerado a respeito do ipv6 é que, embora já esteja sendo adotado gradualmente, ele ainda é um projeto em desenvolvimento. Portanto, alguns

5 Diferente de ipv4, em ipv6 NAT trata da comunicação entre diferentes endereços IP na mesma interface.

conceitos usados hoje poderão muito bem ser alterados amanhã caso não se mostrem adequados ao uso real na internet.

7.6.2.5 - Roteamento

No caso dos módulos processadores possuírem endereços IPs que não estão na mesma sub-rede (isto é, estão em redes diferentes do ponto de vista do inter redes), para que eles possam se comunicar é necessário um *gateway*, que numa tradução livre seria caminho para sair e/ou entrar. O *gateway* irá então rotear esses pacotes de dados de uma rede para outra, que é a função de roteamento.

Os roteadores usam protocolos específicos para a tarefa de roteamento, e mantém tabelas internas que permitem decidir qual rota determinado pacote de dados deverá seguir, no inter redes.

7.6.3 - Camada de transporte

A camada de transporte provê uma comunicação confiável entre dois nós processadores, independente de eles estarem dentro da mesma rede ou não. Essa camada deve garantir que os dados sejam entregues livres de erros, em sequência e sem perdas ou duplicações.

A arquitetura internet especifica dois tipos de protocolos na camada de transporte, o UDP (*User Datagram Protocol*) e o TCP (*Transmission Control Protocol*).

O UDP pode ser considerado como uma extensão do protocolo IP e não oferece nenhuma garantia em relação à entrega dos dados ao destino. Já o protocolo TCP oferece um serviço confiável de transferência de dados, através da implementação de mecanismos de recuperação de dados perdidos, danificados ou recebidos fora de sequência, minimizando o atraso na transmissão.

Para que um mesmo endereço IP possa permitir simultaneamente prover ou acessar diferentes serviços em rede, foi criado o conceito de porta TCP. O número máximo de portas TCP é 65536 (2^{16}), havendo portas para o cliente (usualmente com números acima de 1023) e portas de serviços (usualmente entre 1 e 1023). Desse modo, a comunicação em rede envolve, além do conceito de endereçamento, da necessidade da porta TCP onde o serviço "escuta" à espera de seus clientes e da porta do cliente, que será negociada quando este tentar acessar o serviço. Por exemplo, determinado usuário endereça seus pacotes TCP para a porta 80 (porta da WWW) de determinado servidor *web*, mas para efetuar esse acesso e apanhar o conteúdo do *site* esse cliente precisa abrir uma porta na sua máquina local, por exemplo 19732. Criado esse canal virtual (conexão), através das portas cliente e servidor é que serão transportados os dados.

7.6.4 - Camada de aplicação

Nessa camada, do lado cliente temos os aplicativos específicos para acessar os serviços disponíveis no inter redes ou internet, que é o lado do servidor. Por exemplo, para efetuar um acesso FTP (*File Transfer Protocol*), o cliente usa um aplicativo específico que acessa o serviço FTP:

ambos (cliente e servidor) precisam usar o mesmo protocolo da camada de aplicação para poder se comunicar.

As aplicações cliente-servidor interagem com a camada de transporte para enviar ou receber dados, que podem ser através do UDP ou TCP.

Alguns exemplos de protocolos da camada de aplicação são:

- FTP: File Transfer Protocol (porta 21), permite a transferência de arquivos;
- SSH: Secure Shell (porta 22), acesso shell remoto, porém é seguro no sentido de criptografar o conteúdo dos pacotes que trafegam entre cliente e servidor e vice-versa;
- TELNET: Acesso shell remoto (porta 23);
- SMTP: Simple Mail Transfer Protocol (porta 25), protocolo usado para transferência de e-mails;
- DNS: Domain Name System (porta 53), serviço de resolução de nomes;
- HTTP: Hyper Text transfer Protocol (porta 80), serviço *web*;
- POP: Post Office Protocol (porta 110), serviço usado para receber e-mails;
- IMAP: Internet Message Access Protocol (porta 143), serviço usado para receber e-mails, porém mais sofisticado e completo que o POP;
- HTTPS: serviço web seguro (porta 443), criptografa o conteúdo dos pacotes.

Capítulo 8 - Comparação entre TCP/IP e RM-OSI

O modelo OSI (RM-OSI) evoluiu a partir de uma definição formal elaborada por comissões da ISO buscando desenvolver um produto que atendesse não só as necessidades dos usuários, como desenvolvedores de serviços e soluções em rede e fabricantes de equipamentos de redes.

Já o TCP/IP nasceu da necessidade do mercado e de produtos necessários para resolver o problema da demanda por comunicação em rede, e como o seu uso se expandiu rapidamente, uma série de implementações foram feitas para incorporar ao TCP/IP muitos produtos desenvolvidos fora da arquitetura internet.

Usualmente costuma-se dizer que a arquitetura OSI é um modelo de *jure*, enquanto a arquitetura internet é modelo de *fato*. Ou seja, enquanto o OSI academicamente define padrões, a arquitetura internet apresenta produtos ao mercado.

Comparando a estrutura das duas arquiteturas, observa-se que a parte referente às sub-redes de acesso da arquitetura internet corresponde à camada física, enlace e, parcialmente, a de rede no modelo OSI, sem que haja padronização nesse aspecto.

O IP corresponde à camada de rede, enquanto TCP e UDP oferecem serviços semelhantes aos prestados pelos protocolos de transporte do modelo OSI. Já a camada aplicação da arquitetura internet é sozinha responsável pelos serviços prestados pela camada de sessão, apresentação e aplicação do modelo OSI.

Pelo fato da arquitetura TCP/IP possuir menos camadas que o modelo OSI, isso implica na sobrecarga de algumas camadas em funções específicas definidas no modelo OSI. Como exemplo, a transferência de arquivos no ambiente TCP embute as funções correspondentes à camada de apresentação no próprio protocolo FTP. Embora ocorra essa sobrecarga, por outro lado o TCP/IP nos fornece aplicações simples, eficientes e de fácil implementação a nível de produtos.

Já a arquitetura OSI é criticada pelos seus modelos e soluções excessivamente acadêmicos, que atendem a requisições de propósito geral e não facilitam soluções imediatas, em acordo com as exigências dos usuários.

Existe também um esforço de aproximação entre essa duas arquiteturas, tentando aproveitar o que cada uma tem de melhor, buscando-se encontrar uma solução mista.