



FUNDAMENTOS DA COMPUTAÇÃO

PROF. JOSENALDE OLIVEIRA

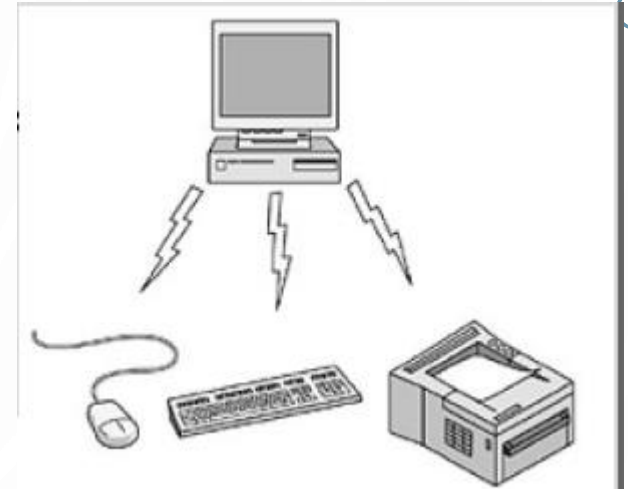
josenalde@eaj.ufrn.br

ANÁLISE E DESENVOLVIMENTO DE SISTEMAS - UFRN

FUNDAMENTOS DE CONECTIVIDADE

Classificação de Redes Quanto ao Alcance geográfico:

PAN (personal area network): proximidade do usuário, alguns metros ($\sim 15\text{m}$), dispositivos portáteis: USB, WPAN (IrDA, bluetooth (BLE), zigbee, uwb...)



USB 3.1 (e USB-C): até 100W: 10 Gb/s (1,2GB/s). Em 1994, versão 1.1: 12 Mb/s.

FUNDAMENTOS DE CONECTIVIDADE

Classificação de Redes:

PAN



Tabela 1 - Comparação entre os Padrões

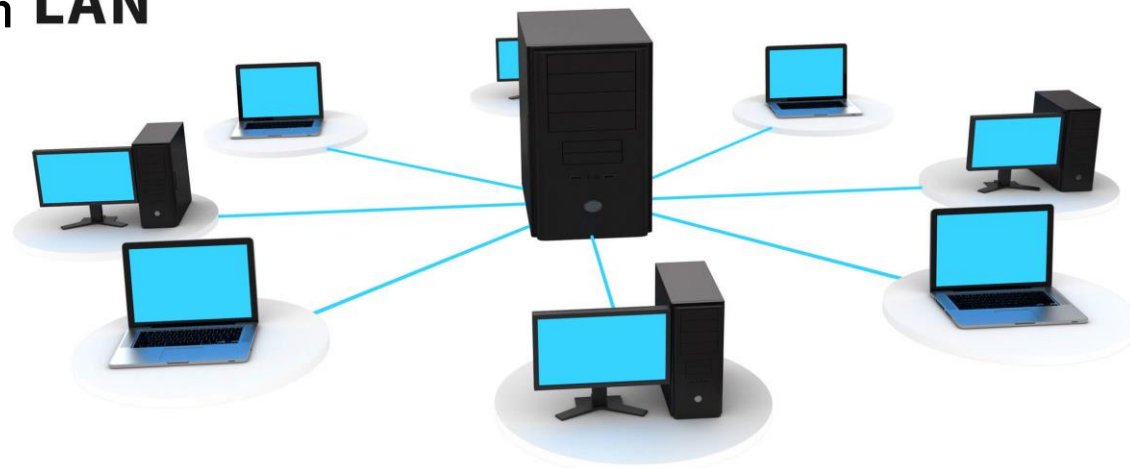
	Bluetooth Clássico	Bluetooth Low Energy (BLE)	ZigBee	Wi-Fi
Padrão de camada física	IEEE 802.15.1	GFSK	IEEE 802.15.4	IEEE 802.11 abg
Frequências	2.4 GHz	2.4 GHz	868 MHz, 915 MHz, 2.4 GHz	2.4 GHz, 5 GHz
Máxima Taxa de Bits (Mbps)	1 até 3	1	0.25	11(b), 54(g), 600(n)
Distância Máxima (m)	10-100	50	10-100	100-250
Consumo de Energia	Alto	Muito Baixo	Muito Baixo	Muito Alto
Vida Útil da Bateria	Dias	Meses a anos	Meses a anos	Horas
Tamanho da Rede (dispositivos)	7	Indefinido	64,000+	255

FUNDAMENTOS DE CONECTIVIDADE

Classificação de Redes:

LAN (local área network): prédios, campus de universidades, 3 km lineares, meios físicos com altas taxas de transmissão, Baixo retardo e poucos erros, “privada”. Pode ser cabeada e/ou sem fios.

Normalmente equipamento de rede ativo, como um **LAN** roteador, gerencia as conexões da LAN e pode prover acesso externo. Um computador também pode atuar como servidor, gateway (“porta”) e gerenciar o tráfego INTERNO-EXTERNO.



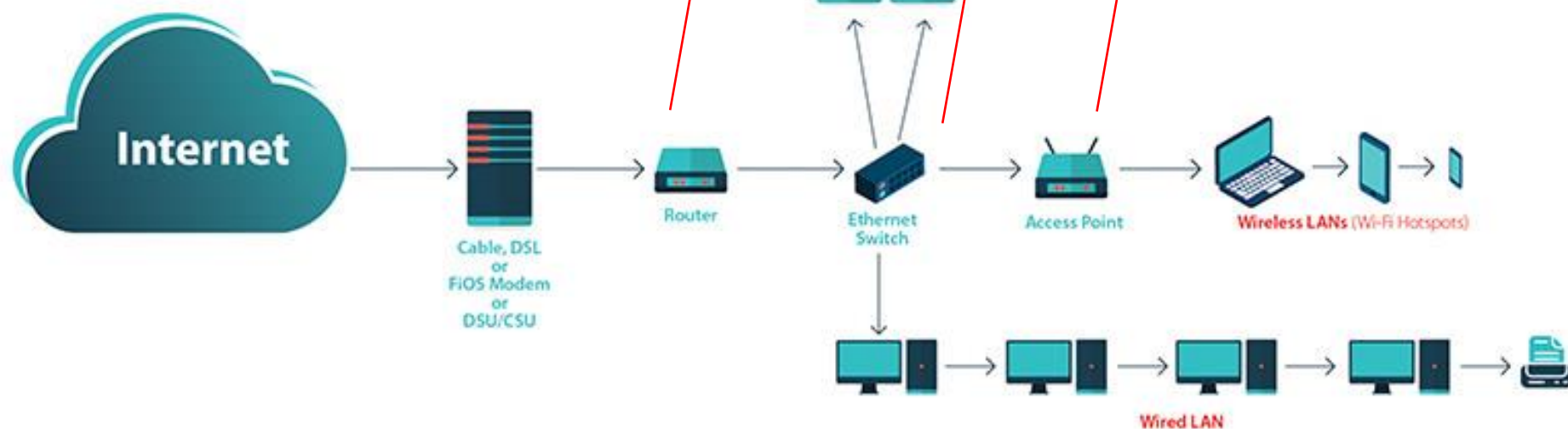
FUNDAMENTOS DE CONECTIVIDADE

Classificação de Redes:

LAN (local área network):

Business Network

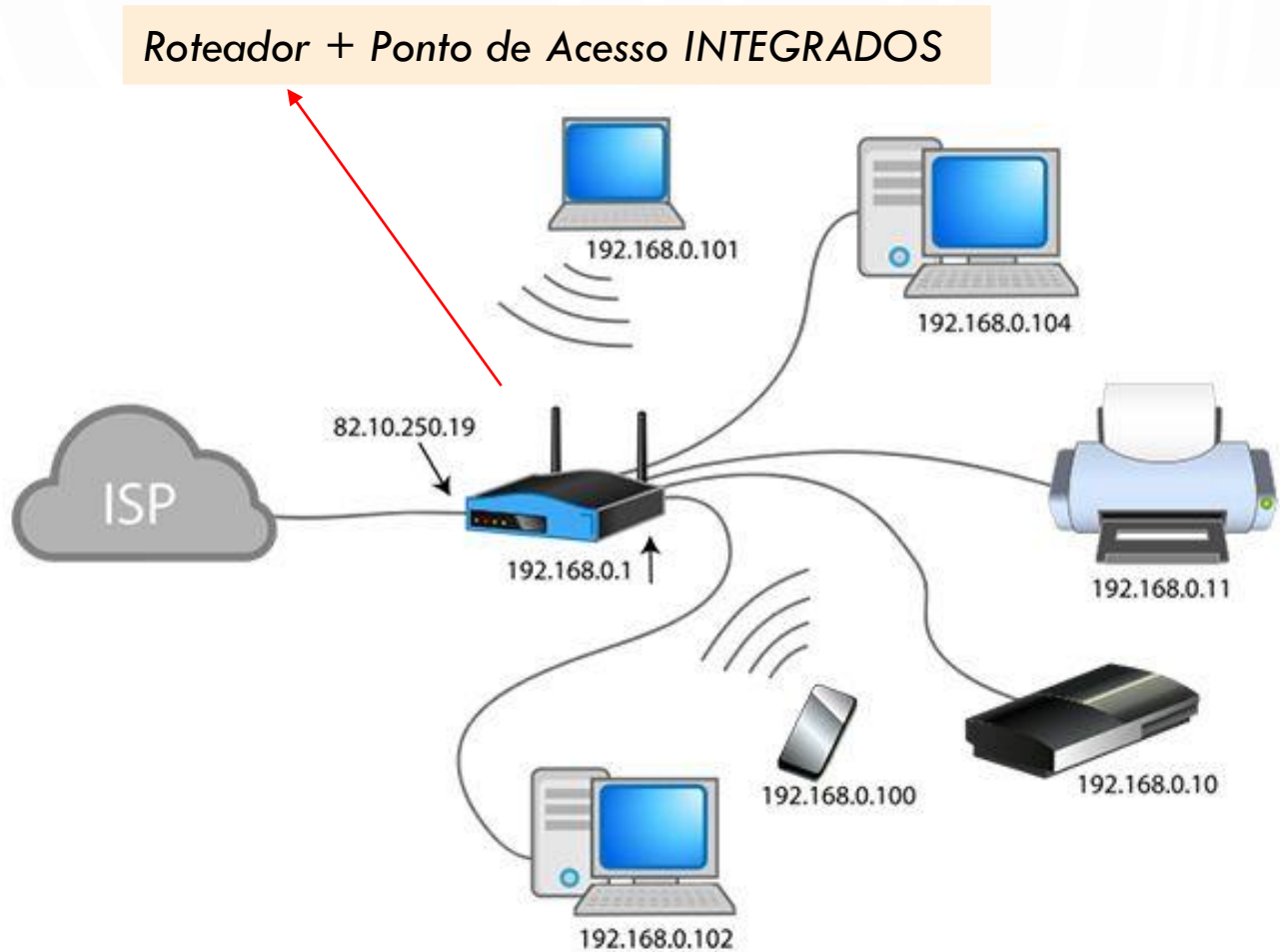
A company local area network (LAN) comprises at the very least a **switch**, **router**, **access point** and **modem**.



FE
FIELDENGINEER.COM

FUNDAMENTOS DE CONECTIVIDADE

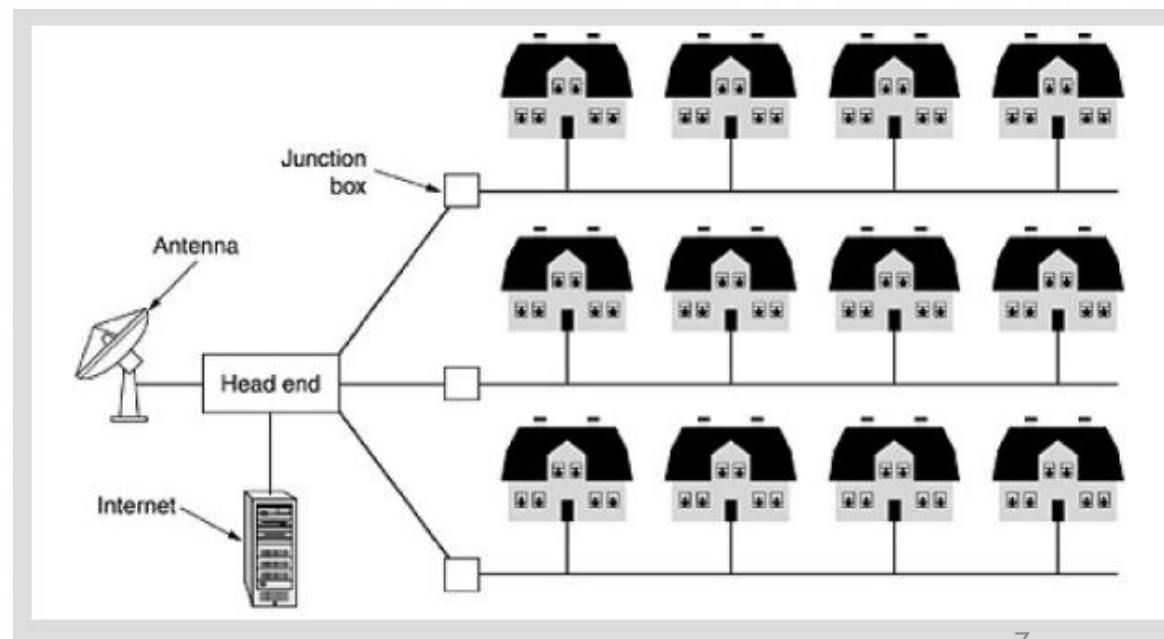
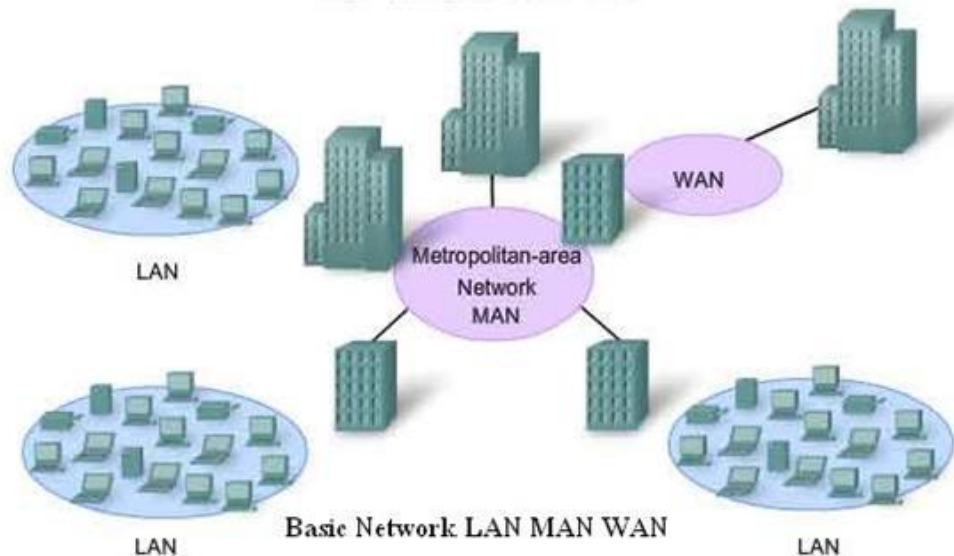
Classificação de Redes:



FUNDAMENTOS DE CONECTIVIDADE

Classificação de Redes:

MAN (metropolitan area network): versão ampliada de LAN, grupos de escritórios vizinhos, cidade inteira, podendo ser pública ou privada. Ex. CATV, cable internet

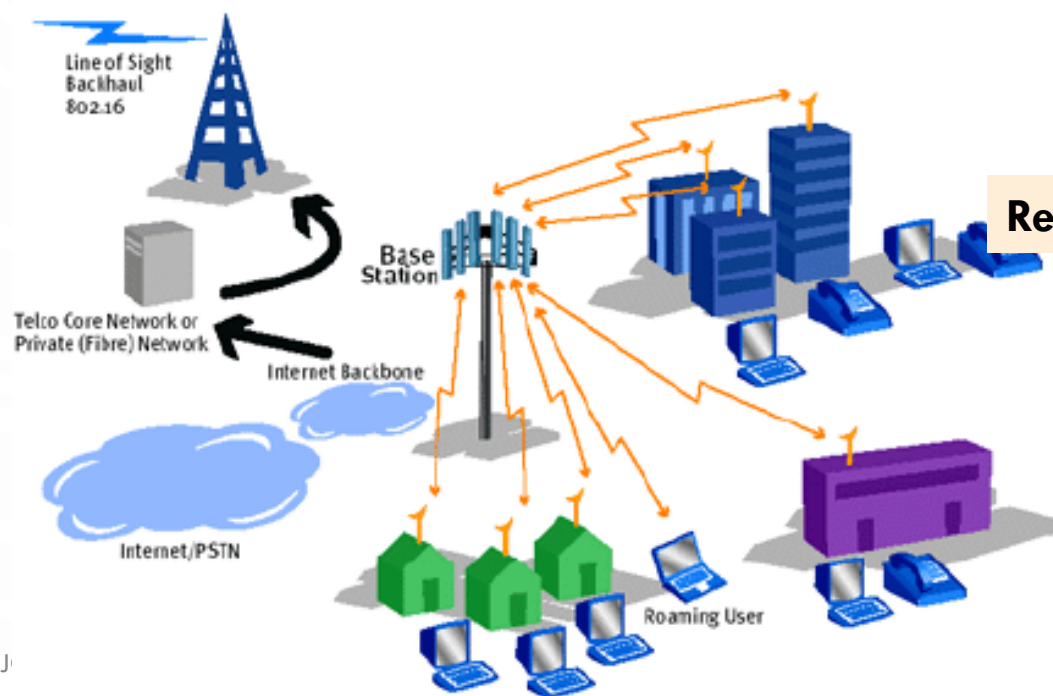


FUNDAMENTOS DE CONECTIVIDADE

Classificação de Redes:

MAN (metropolitan area network): versão ampliada de LAN, grupos de escritórios vizinhos, cidade inteira, podendo ser pública ou privada.

Ex. CATV, cable internet

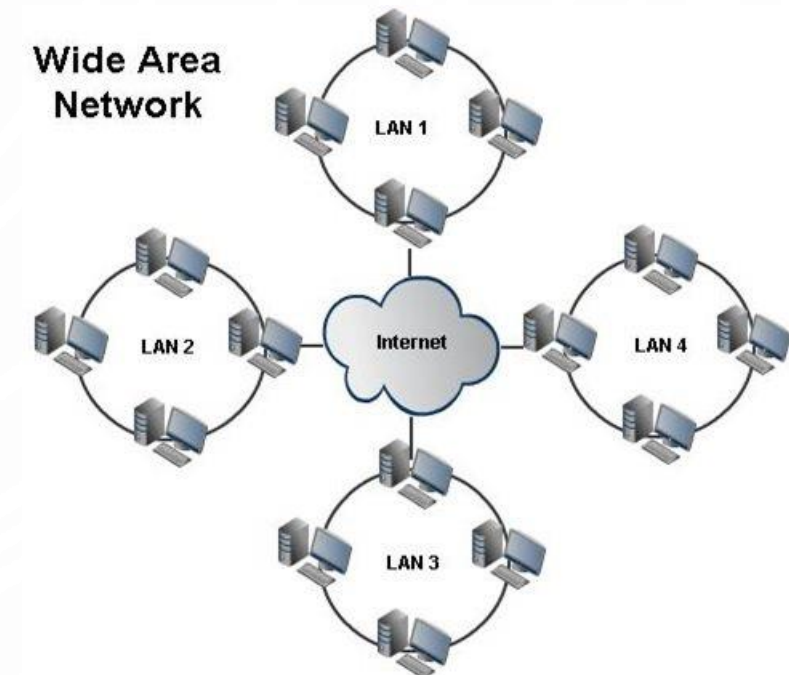
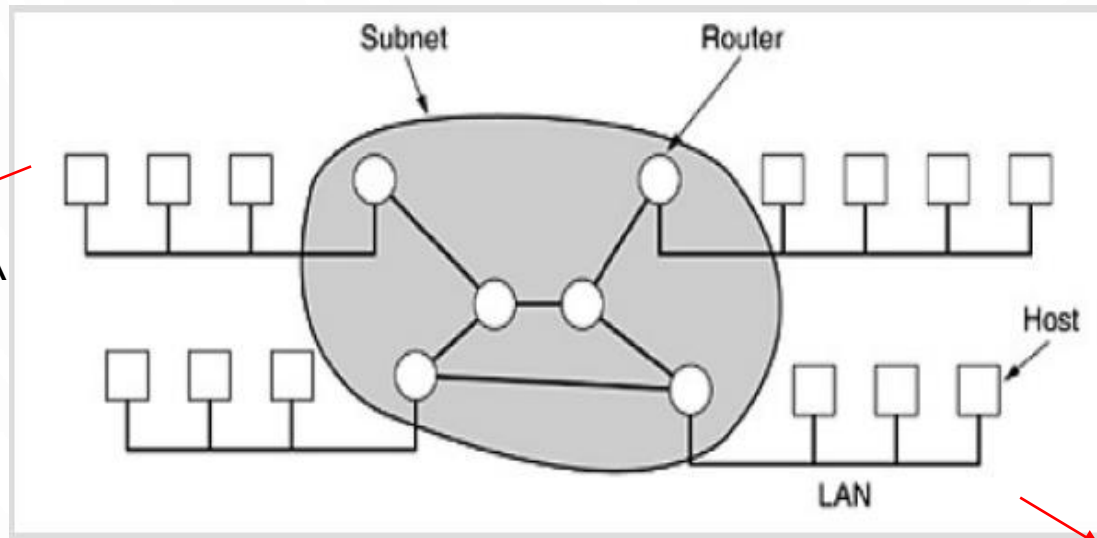


WiMAX



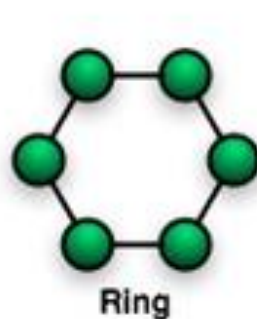
Classificação de Redes:

WAN (wide area network): rede geograficamente Distribuída (países, continentes), com grande infraestrutura (linhas, fibras ópticas, satélites...), as máquinas que executam aplicações do usuário (hosts), conectam-se por sub-redes, que possuem elementos de comutação de pacotes (roteadores)

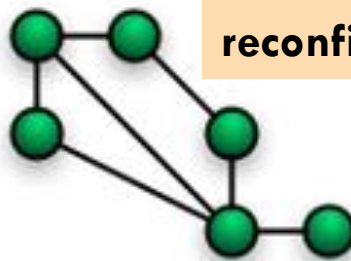


Classificação de Redes Quanto à TOPOLOGIA interna:

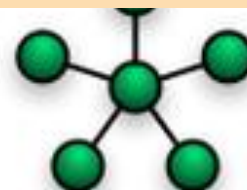
Anel: cada estação verifica se o dado é para si, e, repete, até chegar ao destino. Sofre menos distorção que o barramento. Defeitos individuais, prejudicam a rede.



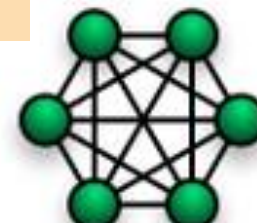
Ring



Mesh



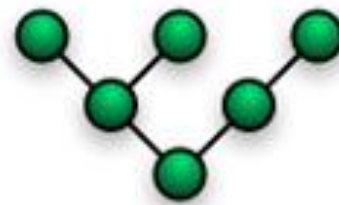
Star



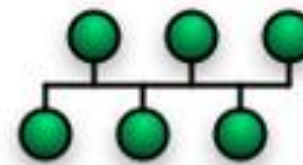
Fully Connected



Line



Tree



Bus

Malha: nós ativos, reconfiguração

Variação da anel

Estrela: as máquinas se conectam a um elemento concentrador (switch), que, **se inteligente**, encaminha à estação destino. Defeitos individuais ou em porta do switch não prejudicam a rede.

Barramento: cada máquina transmite por vez, as demais “escutam” e a destino recebe o dado. Quando uma transmite, o canal fica ocupado. Precisa de TERMINADOR pra indicar as “pontas”

O termo MESH tem sido associado à tecnologia e topologia WI-FI:

- A rede mesh pode ser entendida como um sistema de Wi-Fi inteligente, distribuído de forma automatizada para eliminar as 'zonas mortas' de sinal. Assim, seus dispositivos acabam sendo muito mais do que roteadores e repetidores. Eles formam uma única rede, distribuída em nós (pontos de acesso), onde os usuários podem se conectar a qualquer um deles, sem qualquer distinção e sem perda de sinal ao se afastar do dispositivo principal.
- Para exemplificar: o usuário pode se locomover no espaço entre os diferentes pontos sem perder a conexão ou sequer perceber qualquer mudança, já que há um roteamento automático do sinal. Ou seja, ele pode ir do primeiro andar para o segundo sem sair da rede. O sistema mesh faz essa ligação entre seus nós de forma sincronizada logo que identifica a movimentação. A análise das diversas possibilidades de rotas para o fluxo de informações é instantânea, definindo sempre o melhor caminho – de forma mais rápida e com a menor perda de pacotes – para chegar até o usuário.



Tecnologia Wi-Fi Mesh



Roteador e repetidor comuns

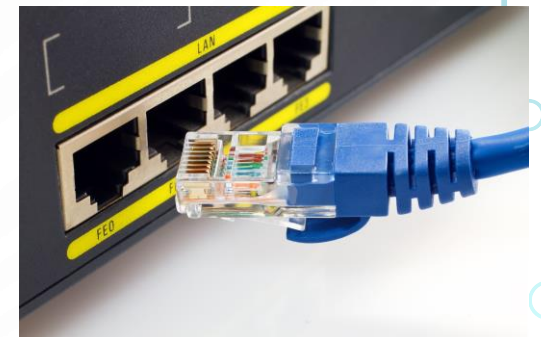
<http://blog.intelbras.com.br/o-que-e-rede-mesh-e-quais-suas-vantagens/>

PROTOCOLOS DE REDE

Para que haja comunicação, TX e RX devem se “entender”, ou seja, conhecer previamente e seguir conjunto de regras (alfabeto, fonemas, vocabulário, gramática etc.). Dois protocolos são muito comuns no estudo de redes e merecem atenção: o ETHERNET, e o TCP/IP.

ETHERNET: tecnologia para LANS baseada no conceito de transmissão de pacotes. IEEE 802.3 – define padrões de cabeamento e sinais elétricos, interfaces, para a camada física e protocolos para a camada de acesso ao meio (MAC – MEDIA ACCESS CONTROL). Cada ponto (nó, interface) tem um endereço de 48 bits.

Usa SWITCH Ethernet que "aprende" quais são as pontas associadas a cada porta, e assim ele para de mandar tráfego broadcast para as demais portas a que o pacote não esteja endereçado, isolando os domínios de colisão. Desse modo, a comutação na Ethernet pode permitir velocidade total de Ethernet no cabeamento a ser usado por um par de portas de um mesmo switch. (**CSMA/CD e CSMA/CA**)



PROTOCOLOS DE REDE

- Velocidades ETHERNET:
- 10 megabits/seg: 10Base-T Ethernet (IEEE 802.3)
- 100 megabits/seg: Fast Ethernet (IEEE 802.3u)
- 1 gigabits/seg: Gigabit Ethernet (IEEE 802.3z) 1 GbE
- 10 gigabits/seg: 10 Gigabit Ethernet (IEEE 802.3ae) 10 GbE

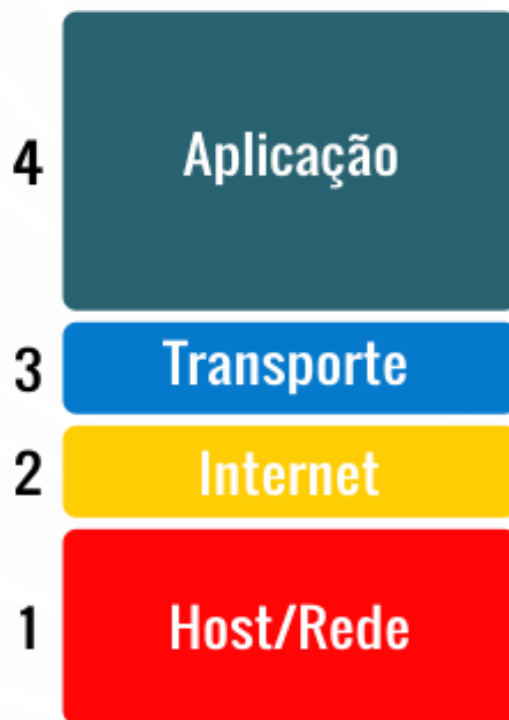


COMUNICAÇÃO EM REDE – CONCEITO DE CAMADAS

Modelos de referência – TCP/IP (“internet”)



Modelo de Referência OSI



Pilha de Protocolos da Internet

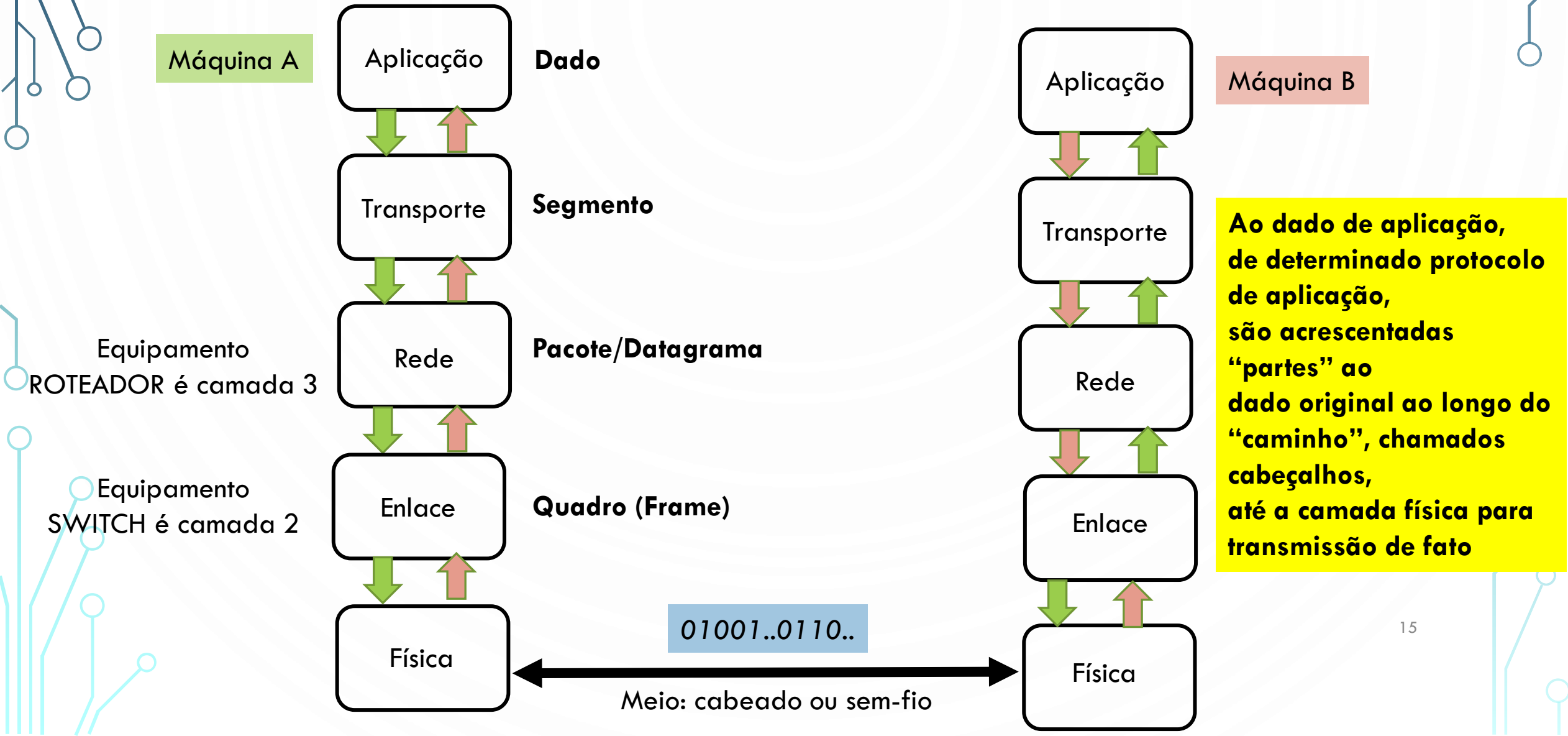


Modelo TCP/IP

} Interface de Rede

↔
EQUIVALENTES

CONCEITO DE CAMADAS - Modelos de referência - TCP/IP ("internet") - PILHA DE PROTOCOLOS



Em termos de programação, pense no conceito de ESTRUTURAS

Campos do segmento TCP

bit	0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24	25	26	27	28	29	30	31				
Porta de origem																	Porta de destino																			
Número de sequência																																				
Número de confirmação																																				
HLEN				Reservado								U	R	G	A	C	K	P	S	H	R	S	T	S	Y	N	F	I	N	Janela						
Soma de verificação																	Ponteiro de urgência																			
Opções (se houver)																								Padding												
Dados																																				
...																																				

```
struct tcpSegmento {  
    short pOrigem;  
    short pDestino;  
    int nSequencia;  
    int nConfirmacao;  
    ...  
    int Dados;  
};
```

OBS: veja no segmento TCP o conceito de PORTA. Segmento TCP é encapsulado em datagrama IP

Para haver comunicação de REDE, assume-se que a ORIGEM sabe o serviço que deseja acessar no DESTINO e quem é o DESTINO, ou seja, numa LAN, qual seria o endereço IP ou nome da máquina (hostname). Num contexto de navegação WEB, sabe-se o nome (URL) de onde se deseja ir, e este é mapeado para um endereço IP (DNS)

OBS: você mandaria uma carta sem saber o destinatário numa comunicação ponto-a-ponto (unicast)? Ou para um grupo de pessoas (multicast), ou para todas as pessoas (broadcast)...

Em termos de programação, pense no conceito de ESTRUTURAS

O comando `hostname` permite obter o nome da máquina

A camada de REDE também incorpora um protocolo simples, apenas para testes de conexão, chamado ICMP. Um comando que usa o protocolo ICMP é o `ping`.

Por exemplo: suponha que desejo “pingar”, ou seja, saber se uma máquina está disponível, basta digitar

`ping <endIpDestino> ou ping <hostnameDestino>`

O endereço IP pode ser mapeado para o nome da máquina por

`ping -a <end. IP. do destino>`

Para haver comunicação de REDE, assume-se que a ORIGEM sabe o serviço que deseja acessar no DESTINO e quem é o DESTINO, ou seja, numa LAN, qual seria o endereço IP ou nome da máquina (hostname). Num contexto de navegação WEB, sabe-se o nome (URL) de onde se deseja ir, e este é mapeado para um endereço IP (DNS)

OBS: você mandaria uma carta sem saber o destinatário numa comunicação ponto-a-ponto (unicast)? Ou para um grupo de pessoas (multicast), ou para todas as pessoas (broadcast)...

Em termos de programação, pense no conceito de **ESTRUTURAS**

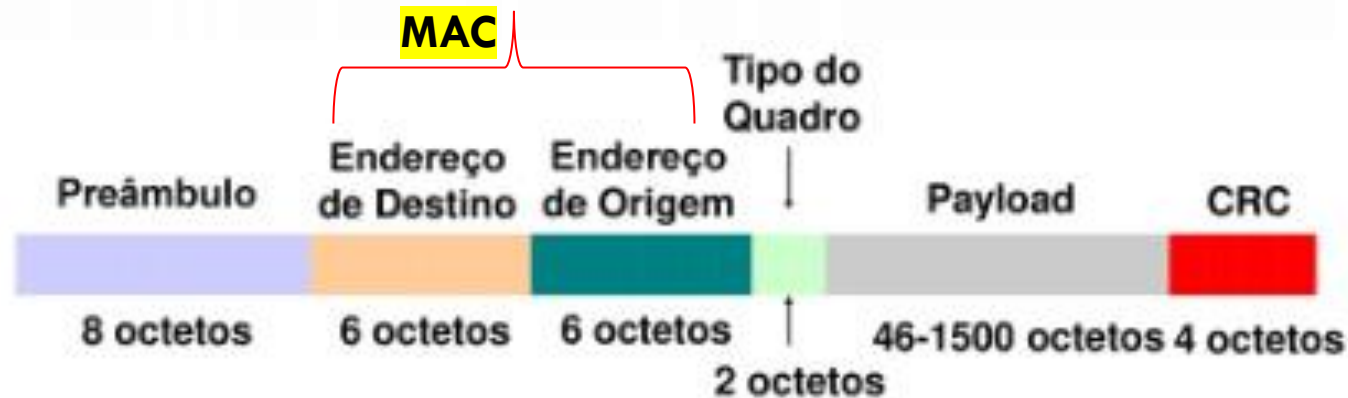
Campos do datagrama IP

0	4	8	16	19	24	31
VERS	HLEN	SERVICE TYPE	TOTAL LENGTH			
IDENTIFICATION			FLAGS	FRAGMENT OFFSET		
TIME TO LIVE		PROTOCOL	HEADER CHECKSUM			
SOURCE IP ADDRESS						
DESTINATION IP ADDRESS						
IP OPTIONS (IF ANY)					PADDING	
DATA						

```
struct ipDatagrama {  
    int ipOrigem;  
    int ipDestino;  
    int Dados;  
    ...  
};
```

OBS: veja no segmento TCP o conceito de **ENDEREÇO IP**. Datagrama IP é encapsulado em **QUADRO ETHERNET**, Na camada de **ENLACE**

Em termos de programação, pense no conceito de ESTRUTURAS



Campos do datagrama IP

```
struct quadroEthernet {  
    long macOrigem;  
    long macDestino;  
    ...;  
    ...  
};
```

➤ O quadro pode variar de tamanho:

- Mínimo: $8 + 6 + 6 + 2 + 46 + 4 = 72$ octetos (60)*
- Máximo: $8 + 6 + 6 + 2 + 1500 + 4 = 1.526$ octetos (1514)*

- * O preâmbulo e o CRC são retirados pelo hardware de rede antes do armazenamento no computador.
- O preâmbulo é um trem de 56 bits, alternados entre 0 e 1 que serve para sincronizar as interfaces de rede + o campo SFD (*Start Frame Delimiter*) = 10101011.

OBS: veja no quadro ETHERNET o conceito de **ENDEREÇO MAC: 48 bits**

Como saber o MAC da máquina destino?

Na prática, existe um protocolo na camada de REDE chamado ARP (Address Resolution Protocol), que obtém o endereço físico (MAC) da máquina DESTINO a partir do seu IP. Para isto, envia pedido em modo Broadcast na REDE.

O contrário também é possível: obter IP a partir do MAC: RARP

SOBRE AO ARP

O comando ARP pode ser usado para obter a tabela de mapeamento IP – MAC gravada na máquina, ou seja, todas as máquinas que a origem tentou conectar, mesmo que a conexão não sucedida, o protocolo consegue obter o MAC associado

O comando `arp` com a opção `-a` retorna a tabela.

Embora o comando ping no endereço de difusão (broadcast) da rede possa ser usado para descobrir os Ips ativos na rede (máquinas ligadas), o SO normalmente tem esta opção desabilitada, e outros softwares são usados, como o NMAP ou net view. Outra opção para descobrir um MAC é tentar usar o getmac com a opção `/s`, ou seja,

`getmac /s <ipDestino>`, sempre com o perfil administrador (root no Linux).

Para haver comunicação de REDE, assume-se que a ORIGEM sabe o serviço que deseja acessar no DESTINO e quem é o DESTINO, ou seja, numa LAN, qual seria o endereço IP ou nome da máquina (hostname). Num contexto de navegação WEB, sabe-se o nome (URL) de onde se deseja ir, e este é mapeado para um endereço IP (DNS)

OBS: você mandaria uma carta sem saber o destinatário numa comunicação ponto-a-ponto (unicast)? Ou para um grupo de pessoas (multicast), ou para todas as pessoas (broadcast)...

COMUNICAÇÃO EM REDE – PROTOCOLOS – EXEMPLO COMPARATIVO

A gravação de dados num arquivo binário obedece a determinada ordem, tipos, tamanhos etc. definida pelo programador e deve ser lido exatamente na mesma ordem de criação, ou seja, segundo uma espécie de “protocolo”

Qualquer alteração na ordem e definições de tamanho no RX, haverá erro na leitura.

Veja este exemplo, em que gravamos num arquivo binário a seguinte sequência:

01 texto de 17 caracteres imprimíveis

01 texto de 5 caracteres imprimíveis

01 variável do tipo *float*

01 vetor de inteiros com 05 valores

01 vetor de inteiros com 02 valores

Logo, precisa ser lido, decodificado, recebido, na mesma ordem e com os mesmos tamanhos e tipos.

```
#include <stdio.h>
#include <stdlib.h>
#include <string.h>

int main() {
    FILE *fp;
    fp = fopen("arq1.bin", "wb");
    if (fp == NULL) {
        printf("Erro ao criar arquivo");
        system("pause");
        exit(1);
    }
    char str[20] = "texto para gravar";
    float x = 5;
    int v[5] = {1,2,3,4,5};
    fwrite(str, sizeof(char), strlen(str), fp);
    fwrite(str, sizeof(char), 5, fp);
    fwrite(&x, sizeof(float), 1, fp);
    fwrite(v, sizeof(int), 5, fp);
    fwrite(v, sizeof(int), 2, fp);
    fclose(fp);
    system("pause");
    return 0;
}
```

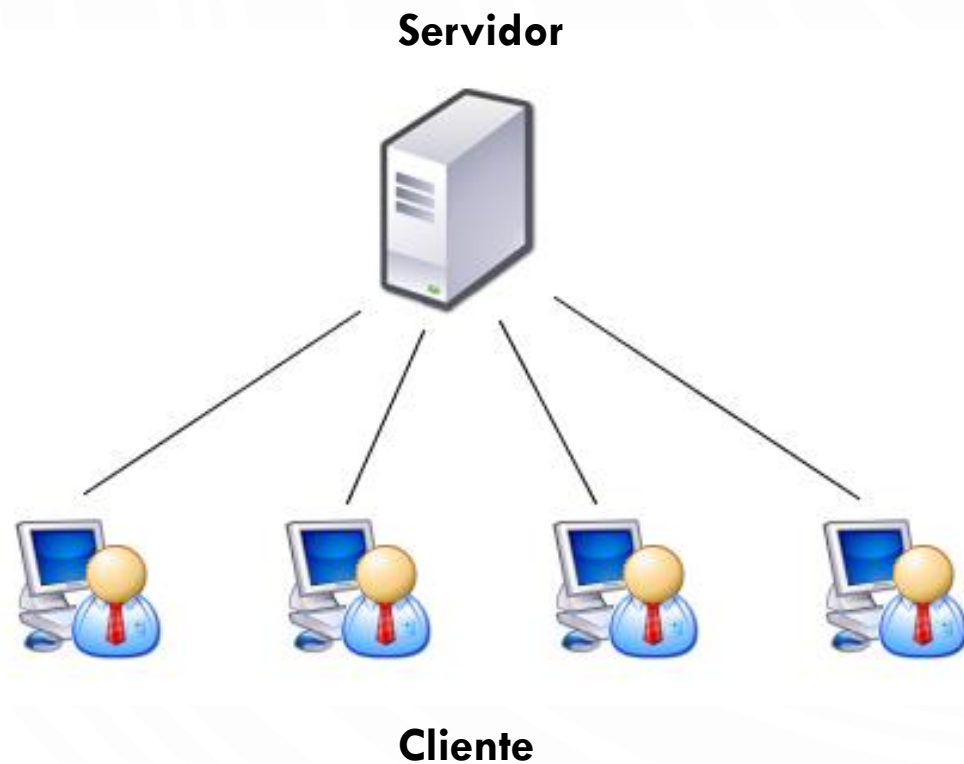
```
int main() {
    FILE *fp;
    fp = fopen("arq1.bin", "rb");
    if (fp == NULL) {
        printf("Erro ao abrir arquivo");
        system("pause");
        exit(1);
    }
    char str1[20], str2[20];
    float x;
    int i, v1[5], v2[2];
    fread(str1, sizeof(char), 17, fp);
    str1[17] = '\0';
    printf("%s\n", str1);
    fread(str2, sizeof(char), 5, fp);
    str2[5] = '\0';
    printf("%s\n", str2);
    fread(&x, sizeof(float), 1, fp);
    printf("%.2f\n", x);
    fread(v1, sizeof(int), 5, fp);
    for (i=0; i<5; i++) printf("v1[%d]=%d\n", i, v1[i]);
    fread(v2, sizeof(int), 2, fp);
    for (i=0; i<2; i++) printf("v2[%d]=%d\n", i, v2[i]);
    fclose(fp);    system("pause");    return 0; }
```

Arquitetura CLIENTE-SERVIDOR: *modelo normalmente usado na internet, em que o servidor (provê serviços de rede, hardware de maior poder computacional) deve estar sempre disponível, à espera da iniciativa do cliente.*

SERVIDOR (response): software que mantém uma porta de comunicação aberta à espera do cliente. Sua localização, seja pelo endereço ou nome, deve ser conhecida por todos os clientes que querem acessá-lo. Pode receber um grande número de solicitações simultâneas. Por causa do alto desempenho, estas máquinas também são conhecidas como SERVIDORES, embora o papel de servidor seja desempenhado por um software.

CLIENTE (request): também é um software, normalmente acionado pelo usuário, razão pela qual é comum que tenha uma interface amigável. Um navegador (browser) web é um exemplo de cliente. Cabe ao cliente iniciar a comunicação com o servidor, seja pelo usuário, seja de forma automática, seja em resposta a um evento ou ação externa. Servidores (software da camada de aplicação) web, de BD, mensagens, e-mails etc. são comuns.

Arquitetura CLIENTE-SERVIDOR:



Exemplo profissional: HPE PROLIANT DL380 Gen10

Intel® Xeon® escalável série 8100/8200 Intel® Xeon® escalável série 6100/6200 Intel® Xeon® escalável série 5100/5200 Intel® Xeon® escalável série 4100/4200 Intel® Xeon® escalável série 3100/3200, 4 a 28 núcleos, 8,25 a 38,50 MB L3, dependendo do modelo de processador, 1 ou 2 processadores, máximo de 3,8 GHz, 24 slots DIMM: 3,0 TB com DDR4 de 128 GB 6,0 TB com módulos de memória persistente de 512 GB HPE DC (disponível em 2H19), LRDIMM de 3,0 TB (24 X 128 GB) Módulos de memória persistente HPE DC de 6,0 TB (12 X 512 GB) - disponível em 2H19, 8 ou 12 SAS/SATA/SSD LFF 8, 10, 16, 18 ou 24 SAS/SATA/SSD SFF 2 SSD SATA M.2 padrão no riser primário, dependendo da configuração 6 unidades traseiras SFF opcionais ou 3 unidades traseiras LFF opcionais e 2 unidades traseiras SFF ou UFF duplas opcionais 20 NVMe SFF opcionais suporte à NVMe via Express Bay limitará a capacidade máxima de unidades, 2 fontes de alimentação, Adaptador HPE 1 Gb 331i Ethernet, 4 portas por controladora e/ou FlexibleLOM opcional, dependendo do modelo

COMUNICAÇÃO EM REDE – PROTOCOLOS

Protocolo: codificação, regras para comunicar, decodificar (ambos os lados precisam conhecer)

Aplicação: comunicação entre os programas de aplicação e os protocolos de transporte da camada abaixo (TCP – TRANSMISSION CONTROL PROTOCOL e UDP – USER DATAGRAM PROTOCOL). Esta comunicação se dá pelo conceito de PORTAS. Os protocolos e portas mais conhecidas são HTTP (hypertext transfer protocol, 80), HTTPS (443), SSL (22), SMTP (simple-mail transfer protocol, 25), FTP (file transfer protocol, 20/21), SNTP (simple network protocol), DNS (domain name system, 53) e Telnet (23), POP, IMAP, DHCP... Ao receber o pacote, o TCP sabe identificar pela porta, a qual protocolo de aplicação deve entregar o dado, após reorganizá-los.

COMUNICAÇÃO EM REDE – PROTOCOLOS

Transporte: entrega fluxo de dados às aplicações, ou seja, chaveia os diversos pacotes de dados que chegam ao computador para as diversas aplicações que fazem acesso à rede. Além disso, pode agregar outras funções como controle de fluxo e congestionamento, garantia de entrega dos pacotes, incluindo a retransmissão de pacotes perdidos e o reordenamento de pacotes (TCP). Para realizar o chaveamento, usa um número de 16 bits (entre 0 e 65535), chamado porta. Toda aplicação tem um endereço de transporte de origem e um endereço de transporte de destino. O protocolo UDP é mais simples, e apenas chaveia os pacotes para as aplicações. Já o protocolo ICMP é usado apenas em aplicações de teste, como o PING, para verificar se um computador está ativo e acessível.

COMUNICAÇÃO EM REDE – PROTOCOLOS

Rede: responsável pelo endereçamento universal, define endereços de origem e destino dos pacotes (IP, Internet Protocol). Atualmente as versões 4 e 6 convivem, sendo a v4 (32 bits, 4 bilhões de elementos conectados) e a v6 (128 bits)

Na v4 são organizados em 4 octetos (conjuntos de 8 bits), indicando o endereço de sub-rede e o endereço do elemento na sub-rede. O número de bits destinado à sub-rede é variável, sendo indicado por um outro valor, a chamada **máscara de sub-rede**. Cada octeto pode ir de 0 a 255.

Exemplo:

Estão na mesma SUBREDE: se comunicam diretamente sem a necessidade de ROTEAMENTO entre redes

Máquina (Host A):

200.125.23.44 (endIP)

255.255.255.0 (mask = 24 bits)

200.125.23.0 (endereço sub-rede, endSubNet)



Máquina (Host B):

200.135.23.68

255.255.255.0

200.125.23.0

Comandos:

Windows: ipconfig

Linux: ifconfig

ping

Algoritmo: endIP & mask = endSubNet

CLASSES ENDEREÇOS IP (detalhes do ipv6... para disciplina específica)

Classe	Primeiro Octeto	Parte da rede (N) e parte para hosts (H)	Máscara	Nº Redes	Endereços por rede
A	1-127	N.H.H.H	255.0.0.0	126 (2^7-2)	16,777,214 ($2^{24}-2$)
B	128-191	N.N.H.H	255.255.0.0	16,382 ($2^{14}-2$)	65,534 ($2^{16}-2$)
C	192-223	N.N.N.H	255.255.255.0	2,097,150 ($2^{21}-2$)	254 (2^8-2)
D	224-239	Multicast	NA	NA	NA
E	240-255	experimental	NA	NA	NA

Endereços reservados para uso interno privado LAN

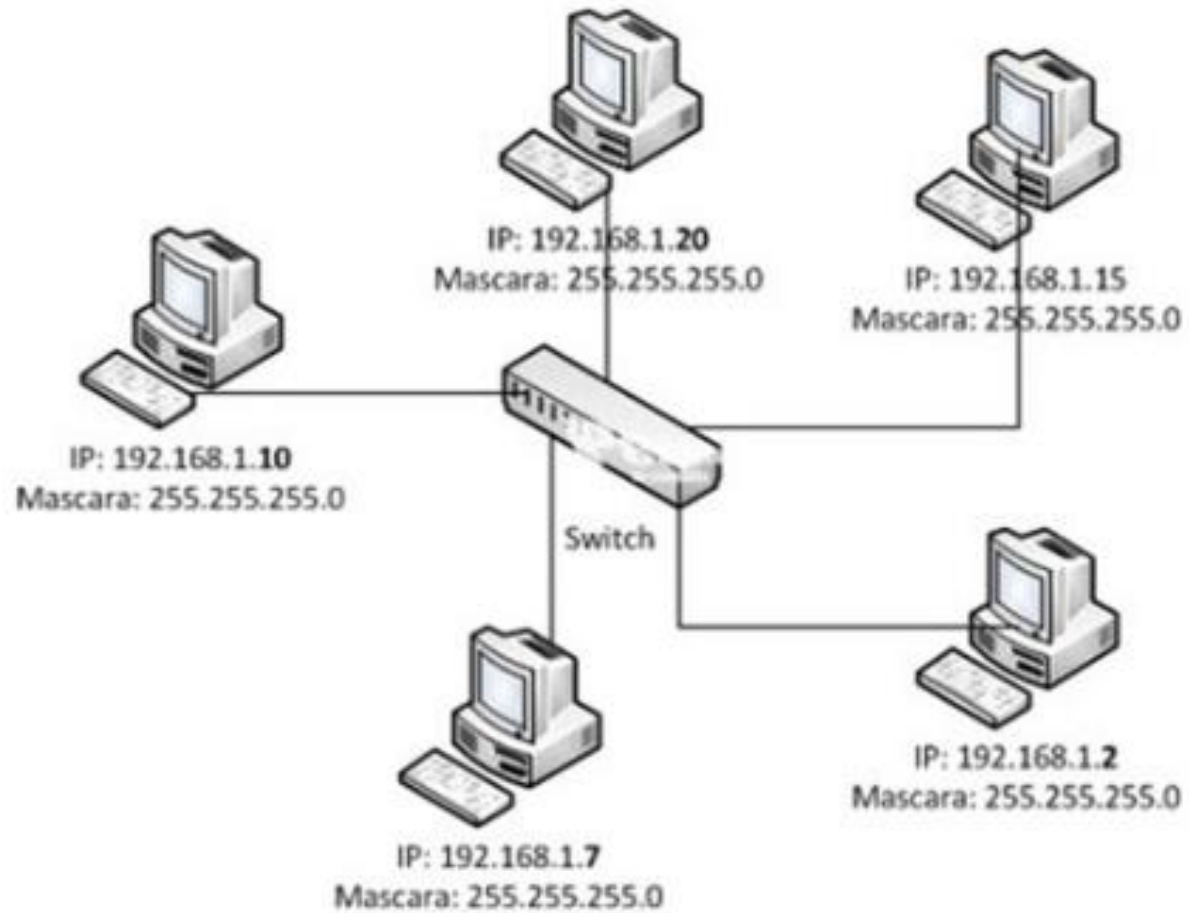
Classe A: 10.0.0.0/8 - 10.0.0.0 – 10.255.255.255

Classe B: 172.16.0.0/12 - 172.16.0.0 – 172.31.255.255

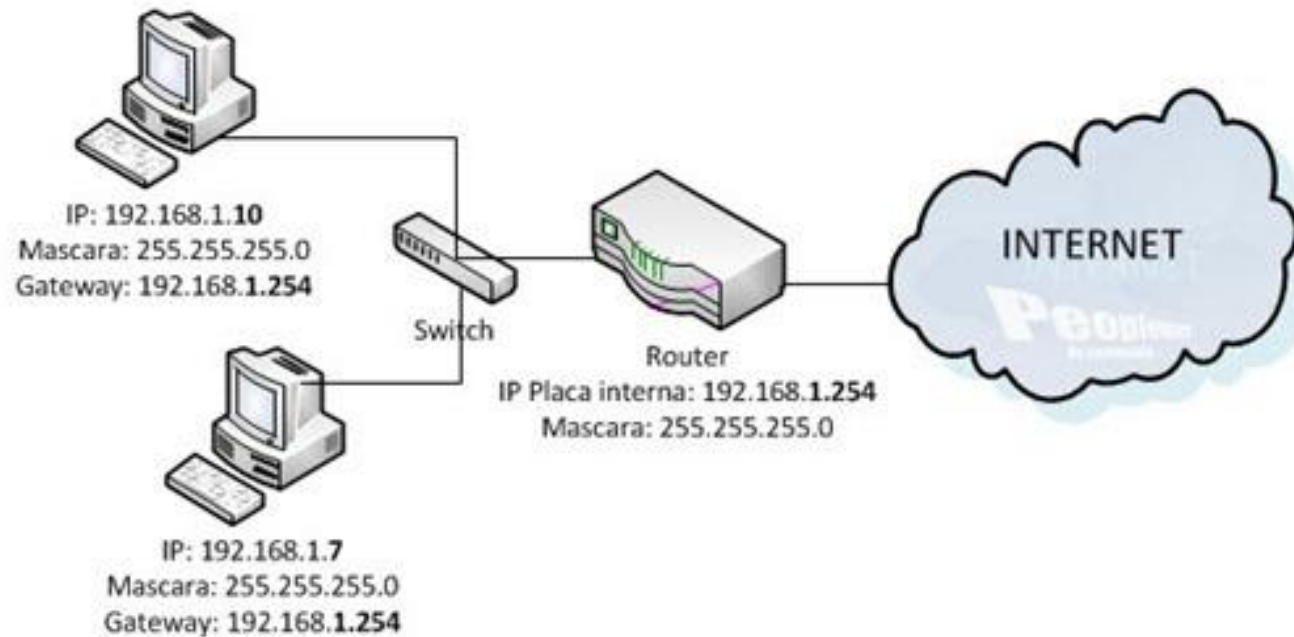
Classe C: 192.168.0.0/16 - 192.168.0.0 – 192.168.255.255

Classe B: 169.254.0.0/16 - 169.254.0.0 – 169.254.255.255 (Windows atribui se não consegue atribuir IP pelo DHCP)

Cenário LAN estrela sem necessidade de GATEWAY, ou seja, ROTEAMENTO IN-OUT



Cenário LAN estrela com necessidade de GATEWAY, ou seja, ROTEAMENTO IN-OUT



Cenário LAN estrela com necessidade de GATEWAY, ou seja, ROTEAMENTO IN-OUT



- Na máquina gateway deve ser criada a rota (tabela de roteamento) para PC1 enxergar PC2 e vice-versa. Isto pode ser feito estaticamente. No Windows, exemplo:

```
route ADD 192.168.1.0 MASK 255.255.255.0 192.168. 0.1 IF 0  
route ADD 192.168.0.0 MASK 255.255.255.0 192.168. 1.0 IF 1
```

- Ou dinamicamente, via protocolos RIP ou OSPF.

Outros protocolos importantes em REDES

SERVIÇO DHCP: dynamic host configuration protocol: máquina que “roda” o serviço (por exemplo, roteador), “escuta” solicitações dos clientes, que estão configurados para obter configurações de rede automaticamente (ip, máscara sub-rede, gateway, dns) e aloca as configurações.

SERVIÇO DNS: serviço de nomes de domínio é responsável por traduzir nomes que são mais fáceis de memorizar, para endereços IP. Assim, nomes como google.com, www.ufrn.br (URL: universal resource locator) podem ser acessados. É preciso que o nome esteja registrado e que um servidor DNS esteja disponível e configurado. O servidor público DNS do Google é 8.8.8.8 (primário) e 8.8.4.4 (secundário).

OBS: às vezes navegamos pelo IP e não pelo NOME: problema provável DNS. Ping não responder pode ser bloqueio de segurança à requisições ICMP.

<https://ipinfo.io/2801:8c:0:b0::a007>

<http://www.ifrn.edu.br> - 200.137.2.130