



UNIVERSIDADE
FEDERAL DO CEARÁ
CAMPUS QUIXADÁ

A Internet e a arquitetura cliente-servidor

QXD0193 - Projetos de Interfaces Web

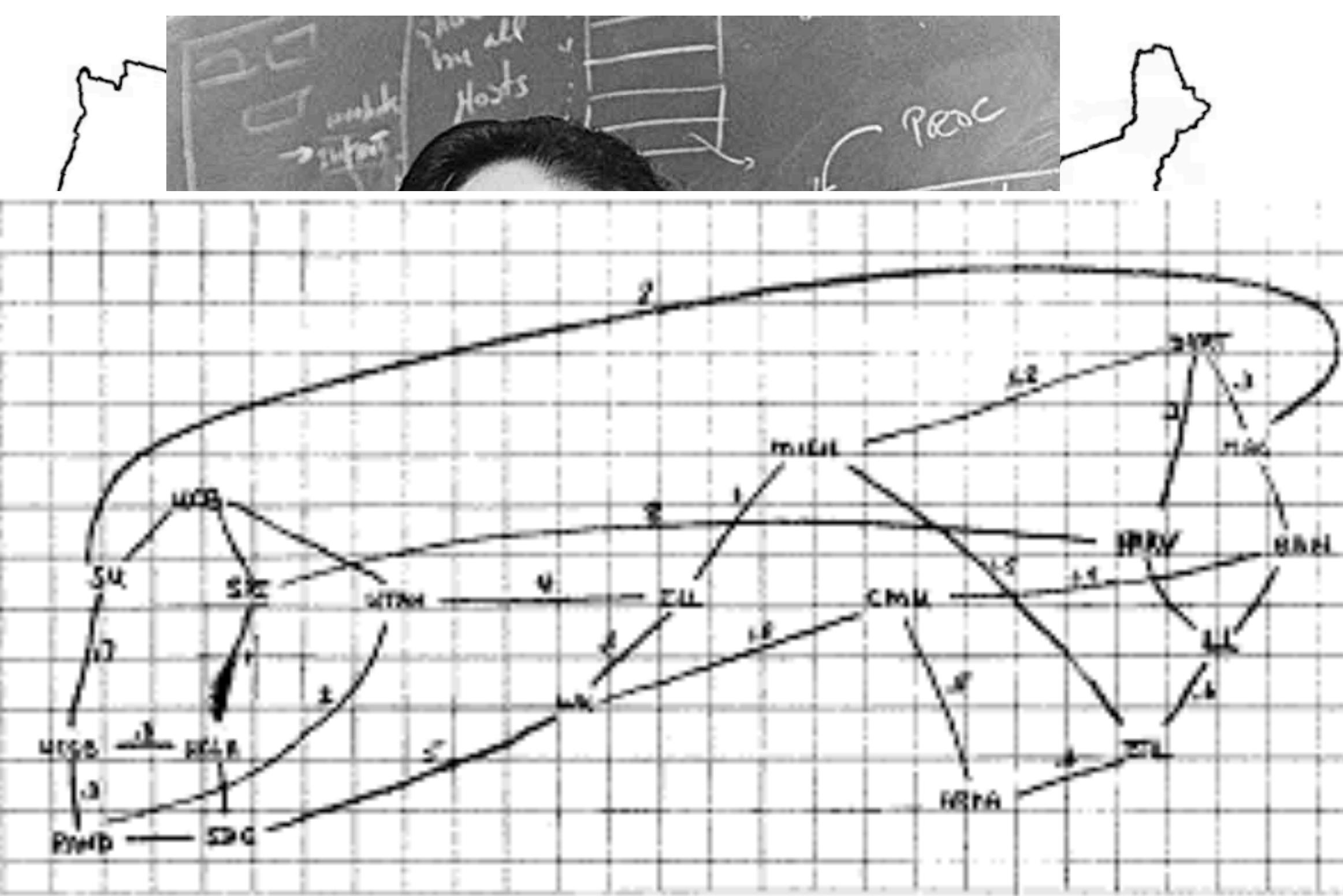
Prof. Bruno Góis Mateus (brunomateus@ufc.br)

Agenda

- A Internet
- Protocolos de comunicação
- A World Wide Web
- Protocolo HTTP

A Internet





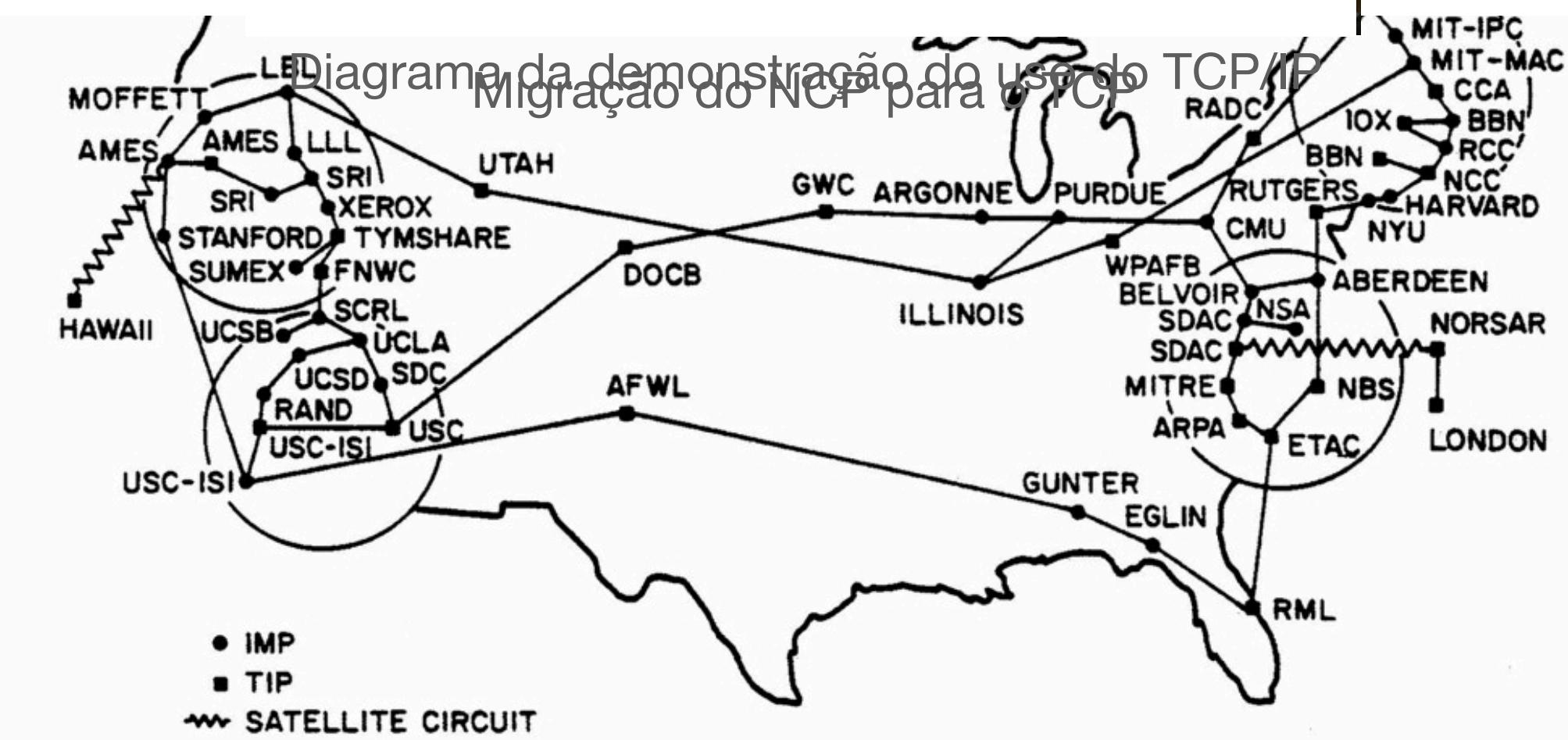
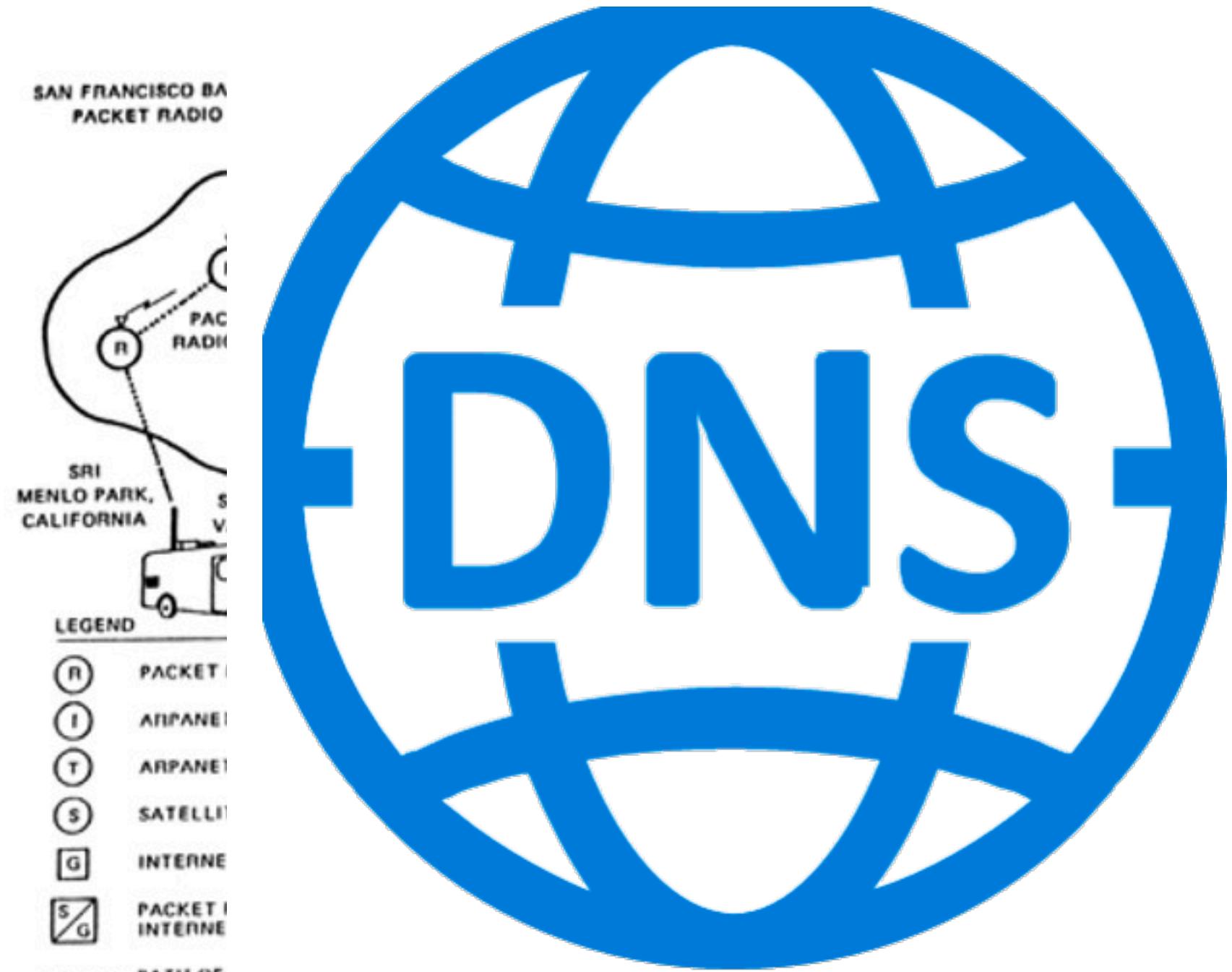
e da ARPA, ou
NET, começa a
scutida



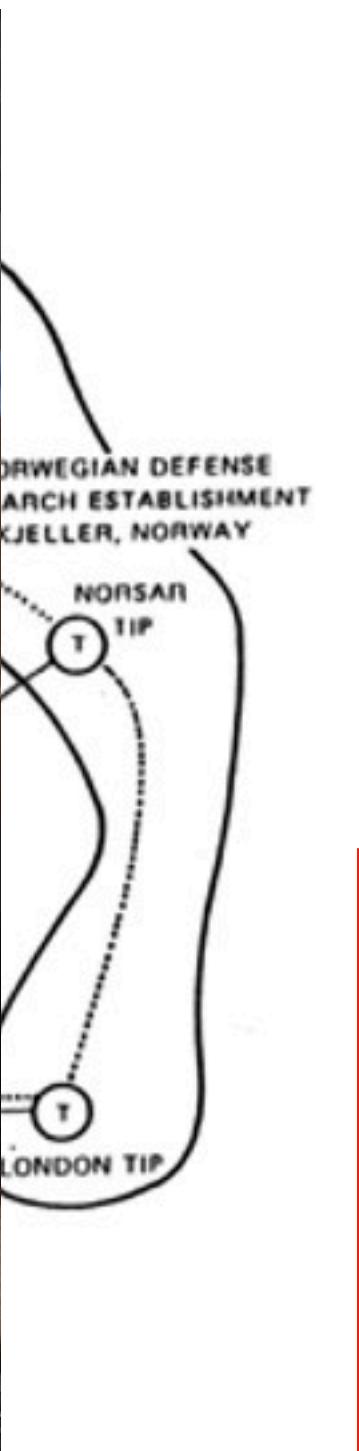
Em 1970, o termo "Internet" é usado pela primeira vez para descrever a rede de computadores que conectava os sistemas de computação das universidades americanas.

1970

O termo Internet é
utilizado pela primeira
vez por Vinton Cerf.



Mapa da ARPANET em 1975



1975
A ARI
considerou
já acuado



24-26 OCTOBER 1972

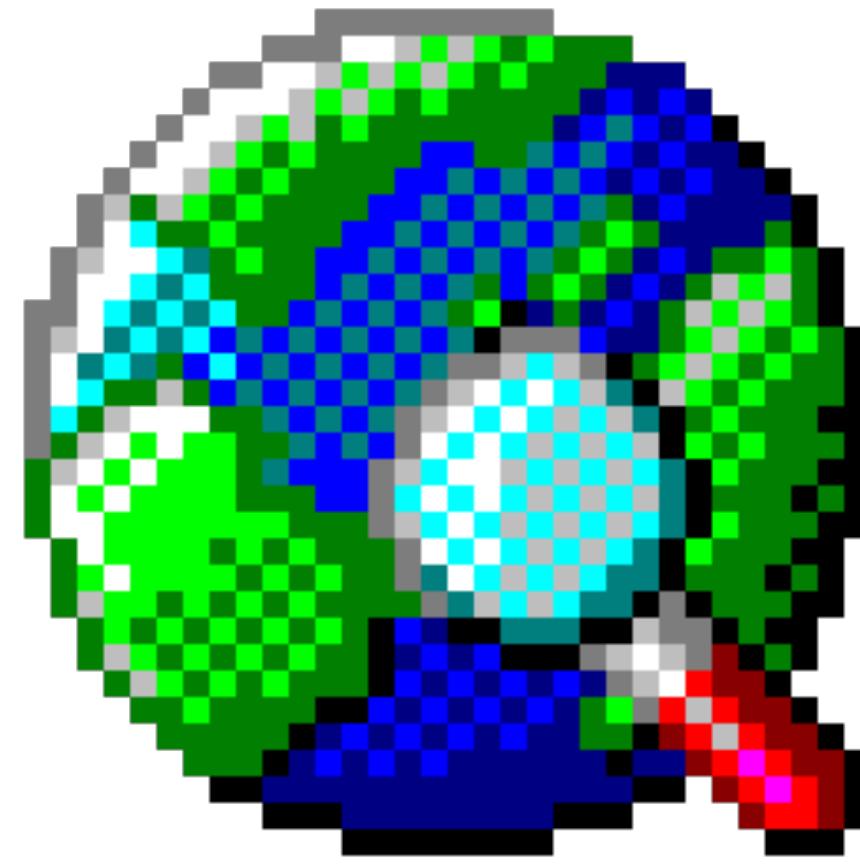
Evento marcado pela criação da ARPANET e sua conexão pública da ARPANET

ão do protocolo

P

1984

O milésimo servidor é instalado
O DNS é estabelecido



Primer navegador a se tornar popular
O Internet Explorer é lançado

Antigo logo do HTML

erface

Object Model

lization

ogy and

ture Initiative

ps

ture

d Multimedia

cessibility
e

rves

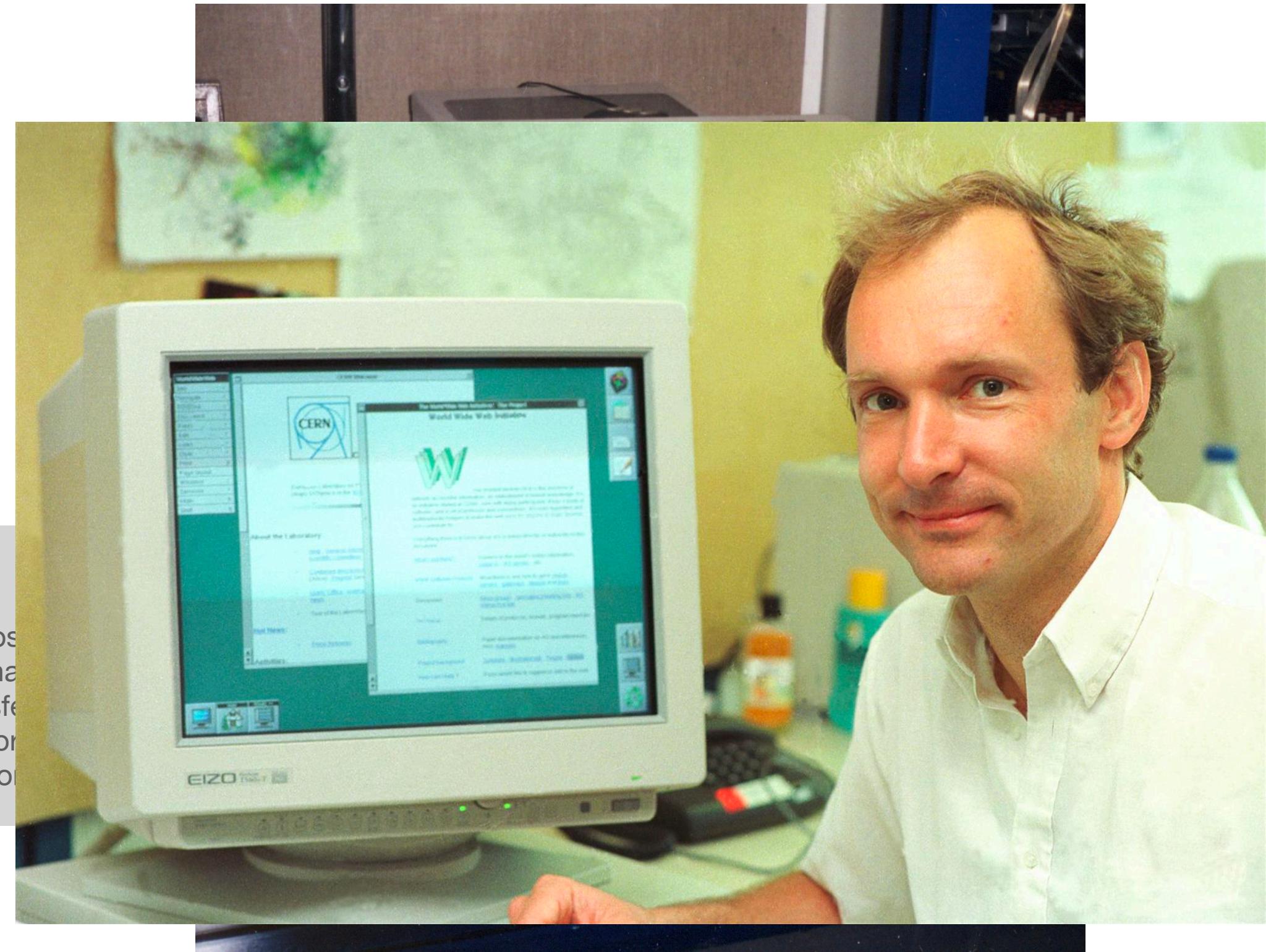
ation Service

s

al

1990

Cria-se: as URLs, ou endereços pra identificar a fonte de página. O HTTP, ou protocolo de transferência de hipertexto; O HTML, que é a forma escolhida pra disposição de conteúdo.



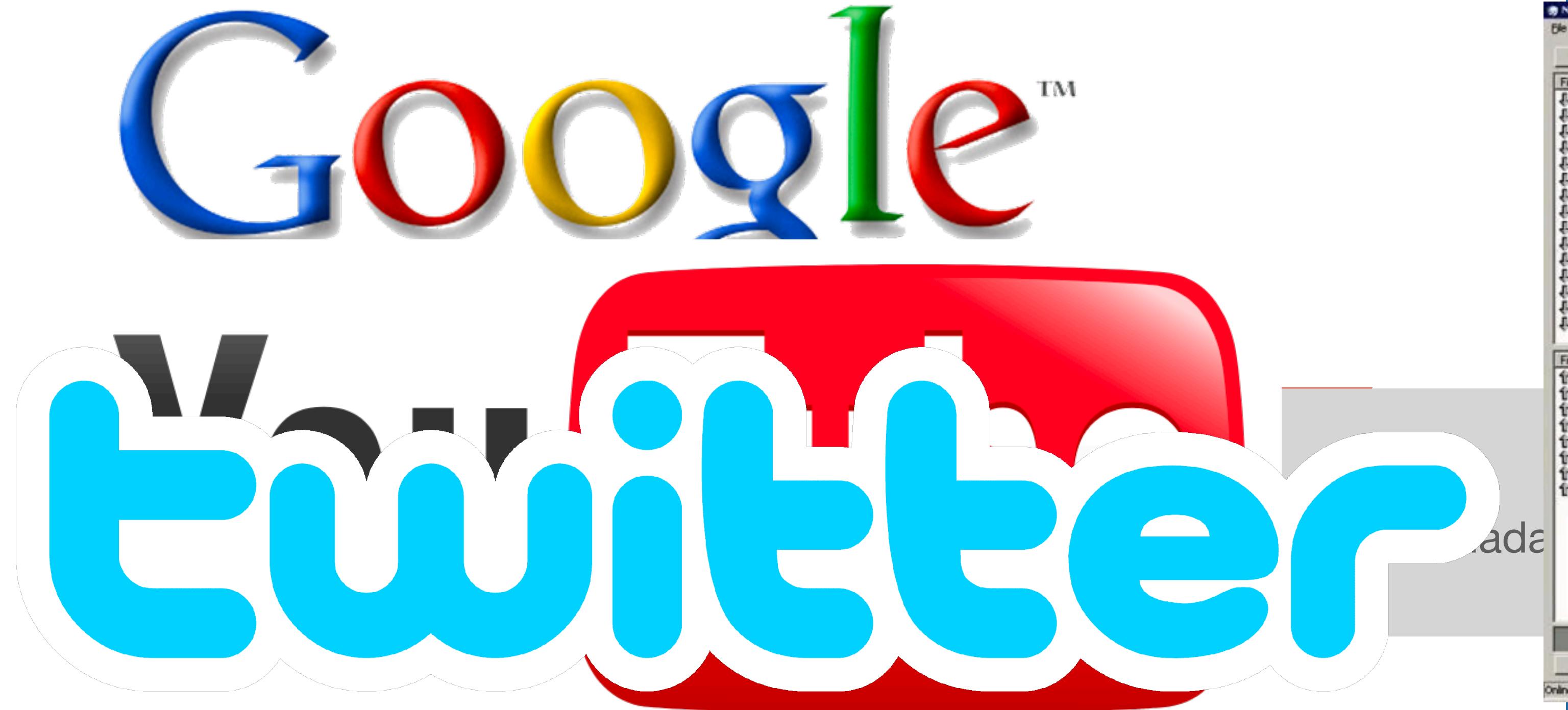
Primeiro servidor de IRC por Urpo Lankinen

1993

Mosaic, o primeiro navegador com informações gráficas é criado

1995

O Internet explorer é lançado.
Amazon é fundada.



[thefacebook]

indefatigable generation of
knowledge, exceeds the short
vehemence of any carnal
pleasure.

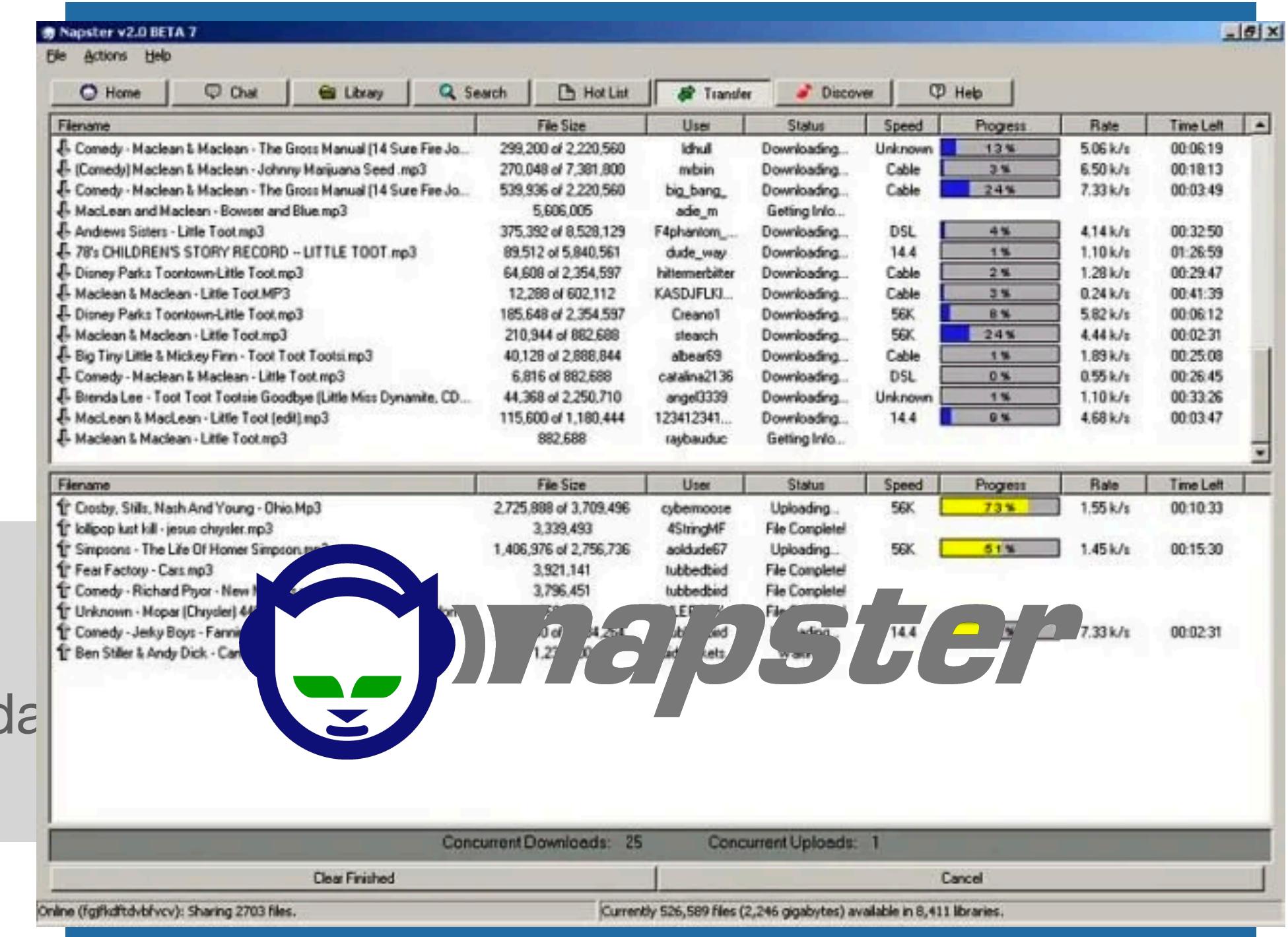
WIKIPEDIA
The Free Encyclopedia

2004

O Gmail é lançado
Facebook é lançado
Orkut é lançado

2006

O Twitter é criado



O ICQ é lançado

A Internet

Aspectos chaves

- Construída sob padrões abertos
- Qualquer pessoa pode utilizar
- As subredes são independentes
- Os computadores podem entrar e sair dinamicamente da rede
- A ausência de um controle centralizado (em grande parte)

A Internet

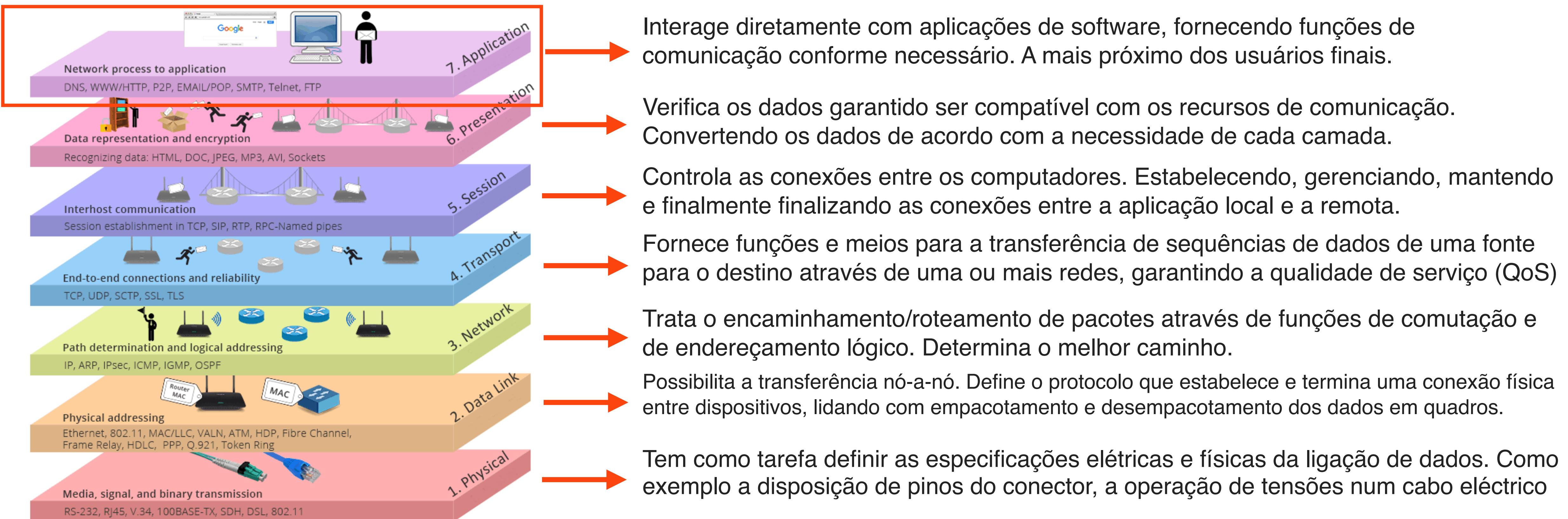
Organizações

- Internet Engineering Task Force – IETF
 - Padrões do Protocolo IP
- Corporation for Assigned Names and Numbers - ICANN
 - Decide sobre o níveis mais altos de domínio
- World Wide Web Consortium - W3C
 - Padrões WEB



A Internet

Modelo de referência OSI



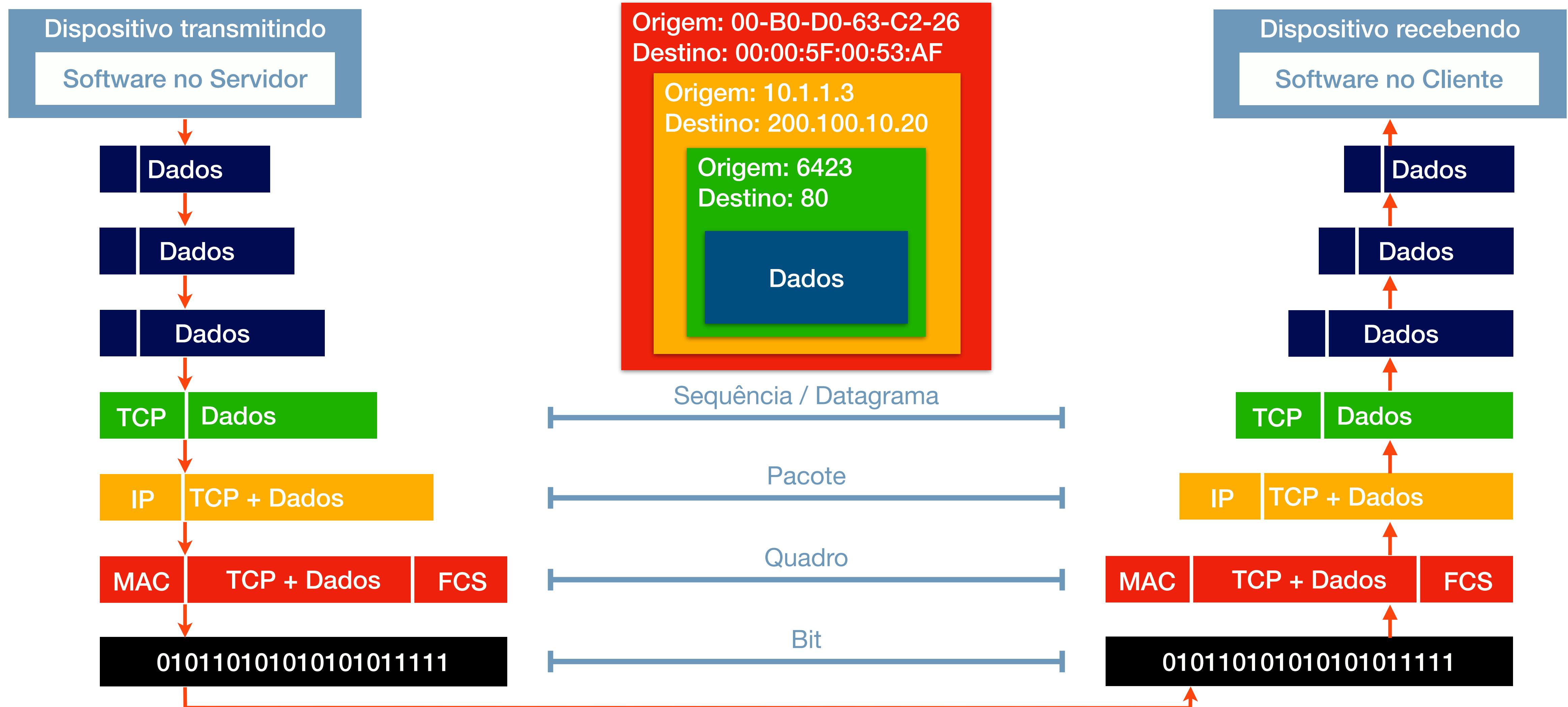
A Internet

Modelo OSI vs TCP/IP

Camadas OSI	Unidade de dados do protocolo	Camadas TCP/IP
Aplicação		
Apresentação	Dados	Aplicação
Sessão		
Transporte	Segmentos (TCP) / Datagrama (UDP)	Transporte
Rede	Pacote	Rede
Enlace	Quadro (<i>Frame</i>)	
Física	Bit	Acesso à rede (Enlace e Física)

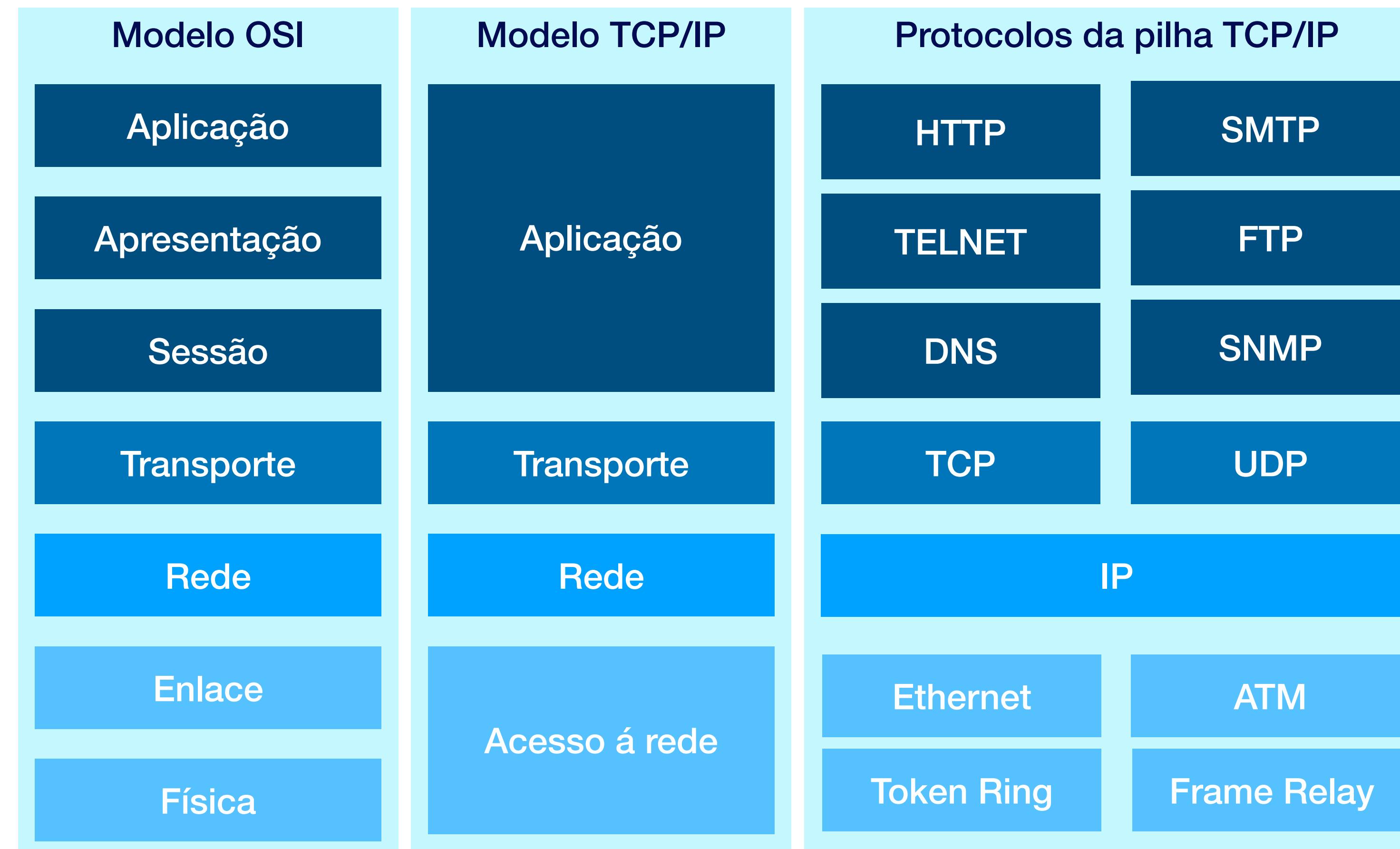
A Internet

Transmissão via TCP/IP



A Internet

Modelo OSI vs TCP/IP



Arquitetura Cliente/Servidor
Modelo computacional onde os servidores fornece, entrega e gerencia os recursos e serviços consumidos pelos clientes

A Internet

Arquitetura Cliente/Servidor

- Clientes
 - Não compartilham recursos com os servidores
 - Software ou aplicação responsável por iniciar a comunicação
 - Geralmente possui uma interface com o usuário
- Servidores
 - Software que recebe as requisições dos clientes e as processa

A Internet

Arquitetura Cliente/Servidor

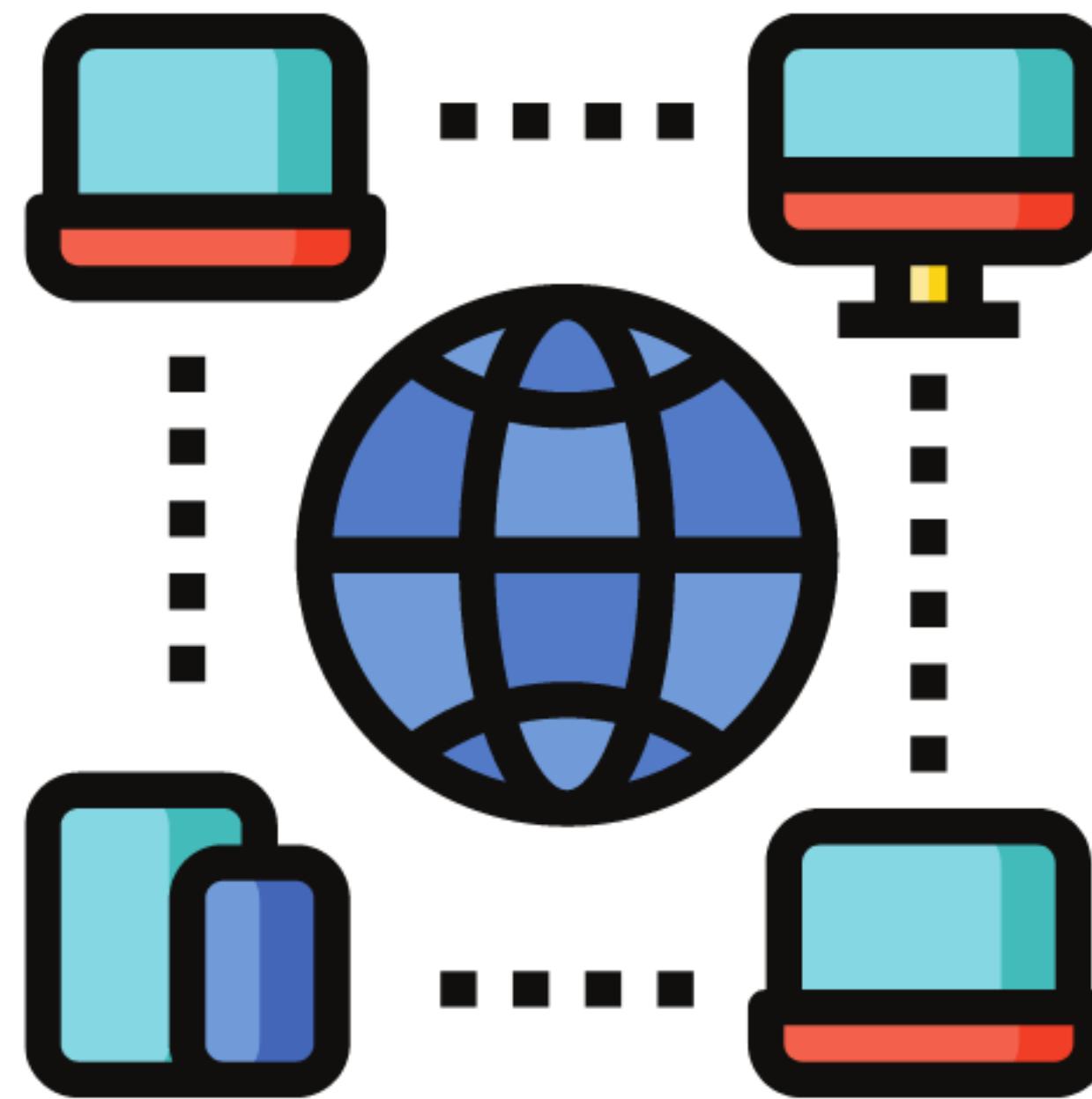
- Características
 - Em geral **clientes e servidores** são bem diferentes em termos de hardware e software
 - **Servidores** podem fornecer diversos serviços simultaneamente
 - Em geral, serviços diferentes são fornecidos por softwares específicos
 - **Clientes e Servidores** interagem diretamente com a **camada de transporte**
 - **Clientes e servidores** precisam da pilha de protocolos completa

A Internet

Arquitetura Cliente/Servidor

- Aplicações reais da arquitetura Cliente/Servidor
 - Servidores de email
 - SMTP (*Simple Mail Transfer Protocol*) + TCP
 - Servidores de arquivo
 - FTP (*File Transfer Protocol*) + TCP
 - Servidores web
 - HTTP (*Hyper Text Transfer Protocol*) + TCP

Protocolos de comunicação



Protocolos de comunicação
“Definem o formato e a ordem das mensagens trocadas entre duas ou mais entidades de comunicação, bem como as ações tomadas na transmissão e/ou recepção de uma mensagem ou outro evento.”

Kurose, 2009

Protocolos de comunicação

- "Regras que governam" a sintaxe, semântica e sincronização da comunicação
- Podem ser implementados pelo hardware, software ou por uma combinação dos dois
- Exemplos:
 - DNS, HTTP - Camada de Aplicação
 - TCP e UDP - Camada de Transporte
 - IP - Camada de Rede

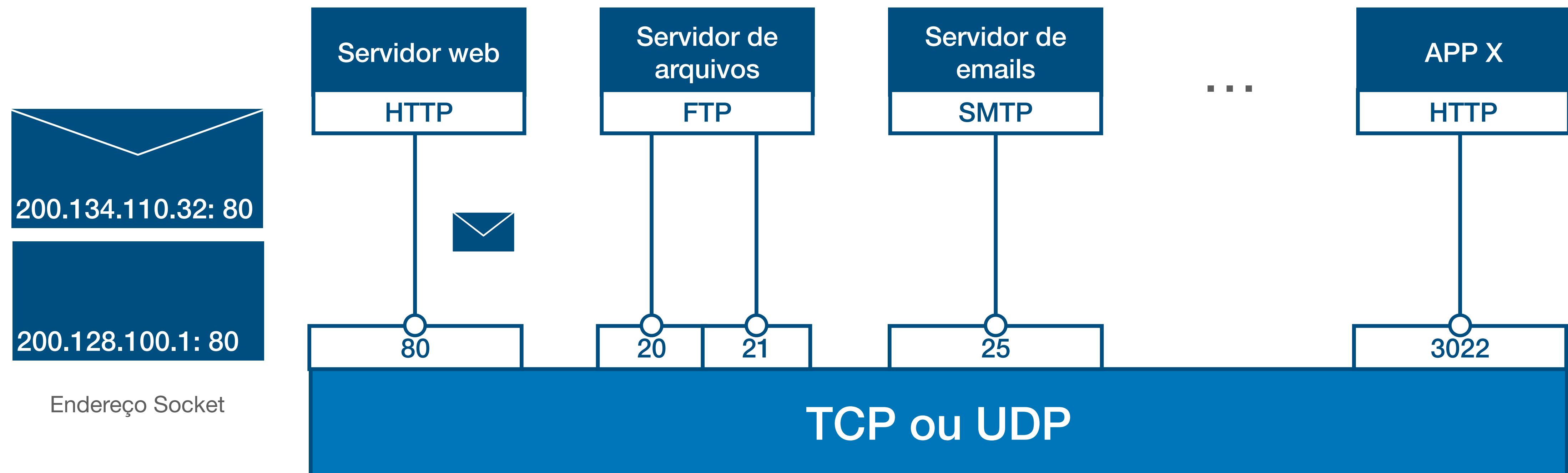
Protocolos de comunicação

Portas

- Permitem que **várias aplicações no mesmo dispositivo físico** se comuniquem com outros dispositivos na rede
 - Comunicação fim-a-fim
 - Identificam os unicamente os processos de origem e destino

Protocolos de comunicação

Portas



Protocolos de comunicação

Camada de transporte: TCP - *Transmission Control Protocol*

- Orientado a conexão
 - Os dispositivos necessitam estabelecer uma conexão antes de iniciar transmissão
 - A conexão deve ser encerrada ao fim da transmissão
- Entrega garantida
 - Garante que os dados enviados sejam entregues na ordem correta
 - Implementa diversos mecanismos de verificação de erros
- Análogo a uma chamada telefônica

Protocolos de comunicação

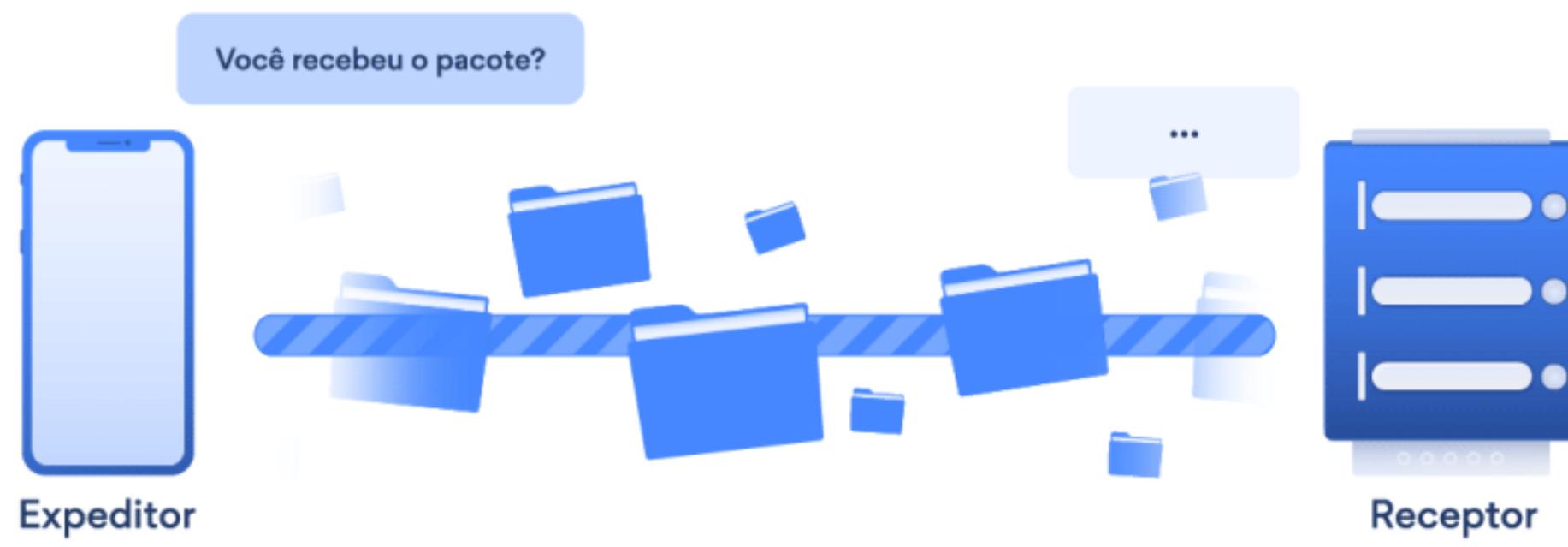
Camada de transporte: UDP - *User Datagram Protocol*

- Comunicação não confiável
 - Entrega não é garantida
- Análogo ao sistema de correio
- Mais rápido que TCP e usado em:
 - Meios de comunicação seguros
 - Para envio de pequena quantidade de dados
 - Serviços tolerantes a pequenas perdas de dados

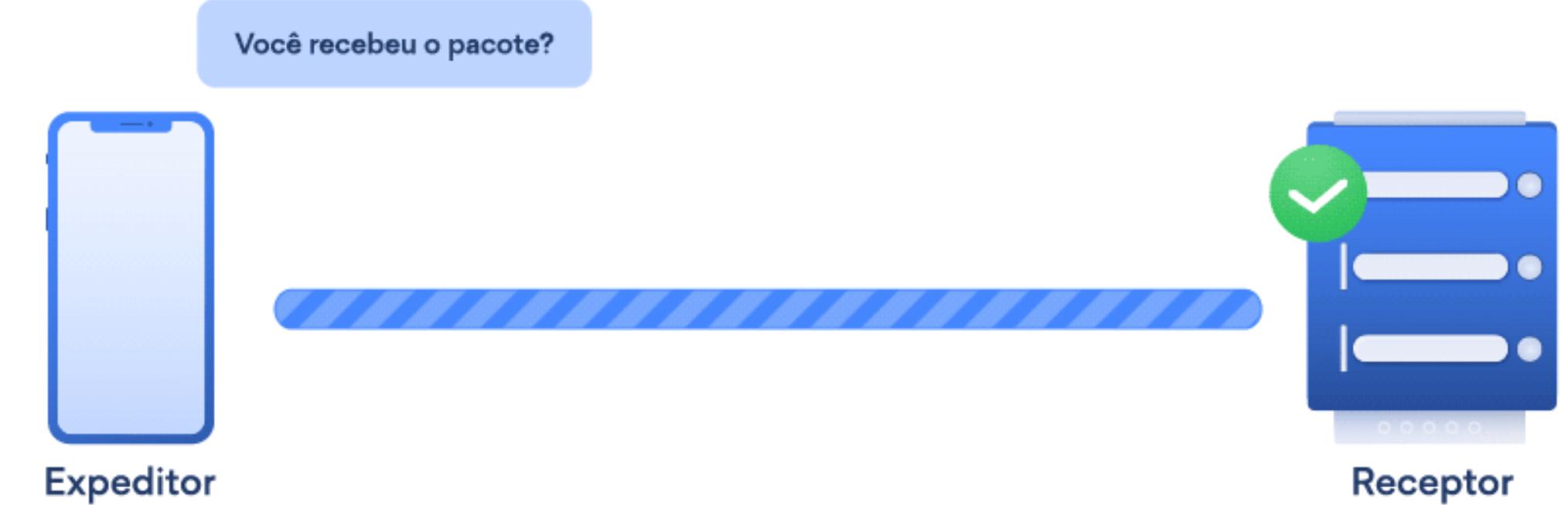
Protocolos de comunicação

UDP vs TCP

Como funciona o UDP



Como funciona o TCP



- DNS - *Domain Name System*
- SNMP - *Simple Network Management Protocol*
- DHCP - *Dynamic Host Configuration Protocol*

- FTP - *File Transfer Protocol*
- HTTP - *HyperText Transfer Protocol*
- SMTP - *Simple Mail Transfer Protocol*

Protocolos de comunicação

Protocolo IP - *Internet Protocol*

- O protocolo que possibilita o envio de dados entre computadores/redes
- Determina as regras que governam o roteamento de pacotes até que eles cheguem ao destino
- Um endereço IP é atribuído a cada interface de rede (wifi, ethernet ...)
 - 32 bits, divididos em 4 números de 8 bits (0-255) para IPv4 para
 - 128 bits, IPv6 devidos em 8 números hexadecimais
 - Identifica unicamente um host da rede

Protocolos de comunicação

Protocolo IP - *Internet Protocol*

IPv4

172 . 16 . 254 . 1

10101100 . 00010000 . 11111110 . 00000001

8 bits

32 bits ou 4 bytes

IPv6

2001:0DB8:AC10:FE01:0000:0000:0000:0000

00100000000001 . 0000110110111000 . 101011000010000 . 11111100000001

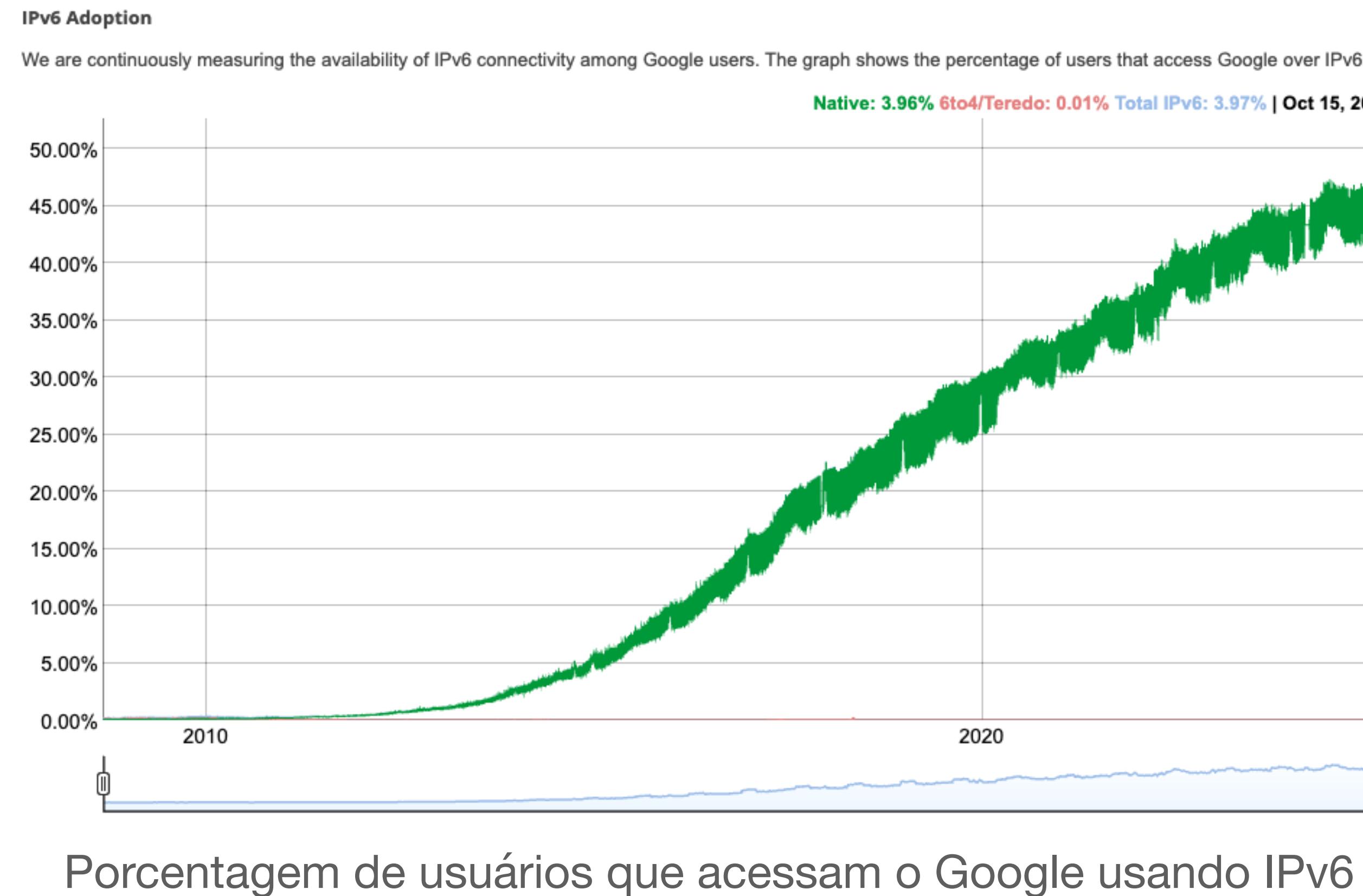
16 bits

128 bits ou 16 bytes

2001:0DB8:AC10:FE01::

Protocolos de comunicação

Protocolo IP - *Internet Protocol*



A World Wide Web



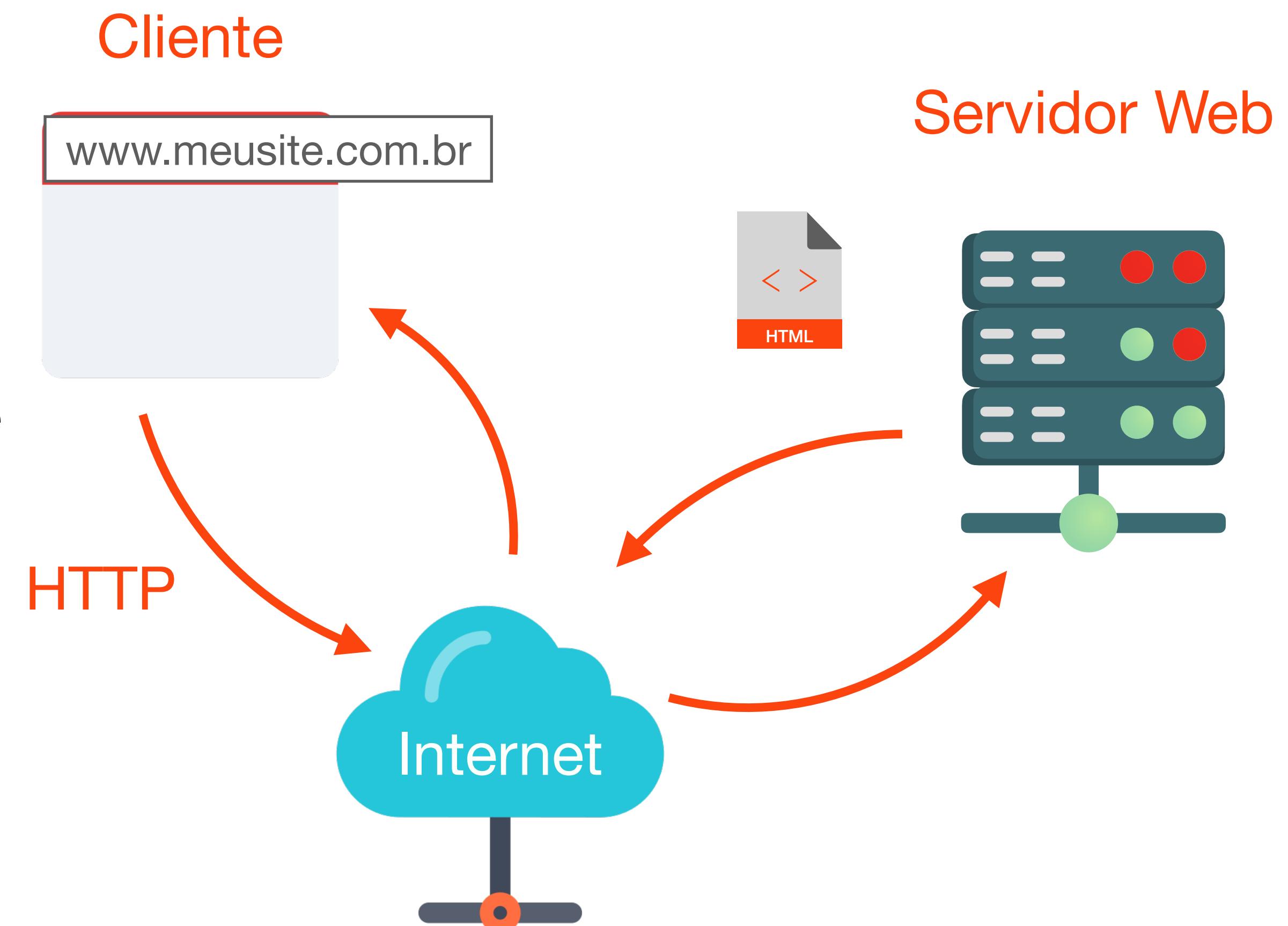
A World Wide Web

- Em 1990 Tim-Berners Lee criou peças chave do que seria a World Wide Web
 - As URLs
 - O HTTP
 - O HTML
- Neste contexto, dois tipos de software ganharam bastante relevância
 - Navegadores
 - Servidores Web

A World Wide Web

- Cenário mais comum:

1. O navegador requisita um recurso ao servidor
2. O servidor web interpreta a requisição
3. O servidor gera uma resposta para o cliente
4. O envia a resposta ao cliente
5. O navegador processa a resposta recebida
6. O navegador apresenta o recurso ao usuário



A World Wide Web

Navegadores

- Navegadores ou *Browsers* são o lado do cliente na arquitetura cliente/servidor da WWW
- Sua principal funcionalidade é apresentar um recurso web solicitado na janela do navegador
 - Usualmente o conteúdo é documento HTML
 - Conversam com servidores web por meio do protocolo HTTP

A World Wide Web

Navegadores

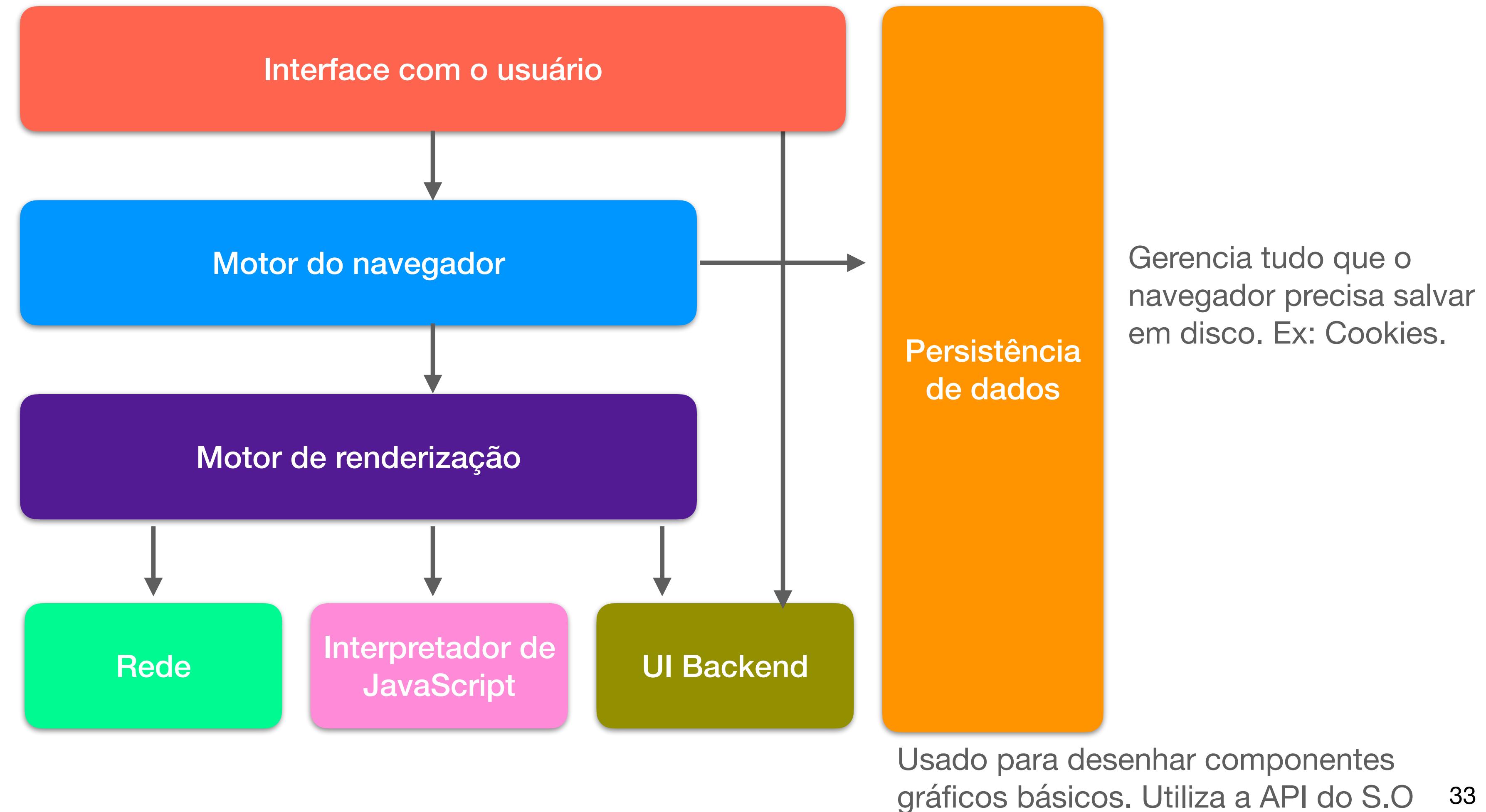
Toda a interface do navegador exceto a porção em que a página requisitada é visualizada.

Camada responsável por interagir o motor de renderização.

Responsável por mostrar o conteúdo requisitado. Ex: HTML & CSS.

Utilizados para realizar operações em rede. Ex: Requisições HTTP.

Responsável pelo parse e execução de código JavaScript



Gerencia tudo que o navegador precisa salvar em disco. Ex: Cookies.

Persistência de dados

Rede

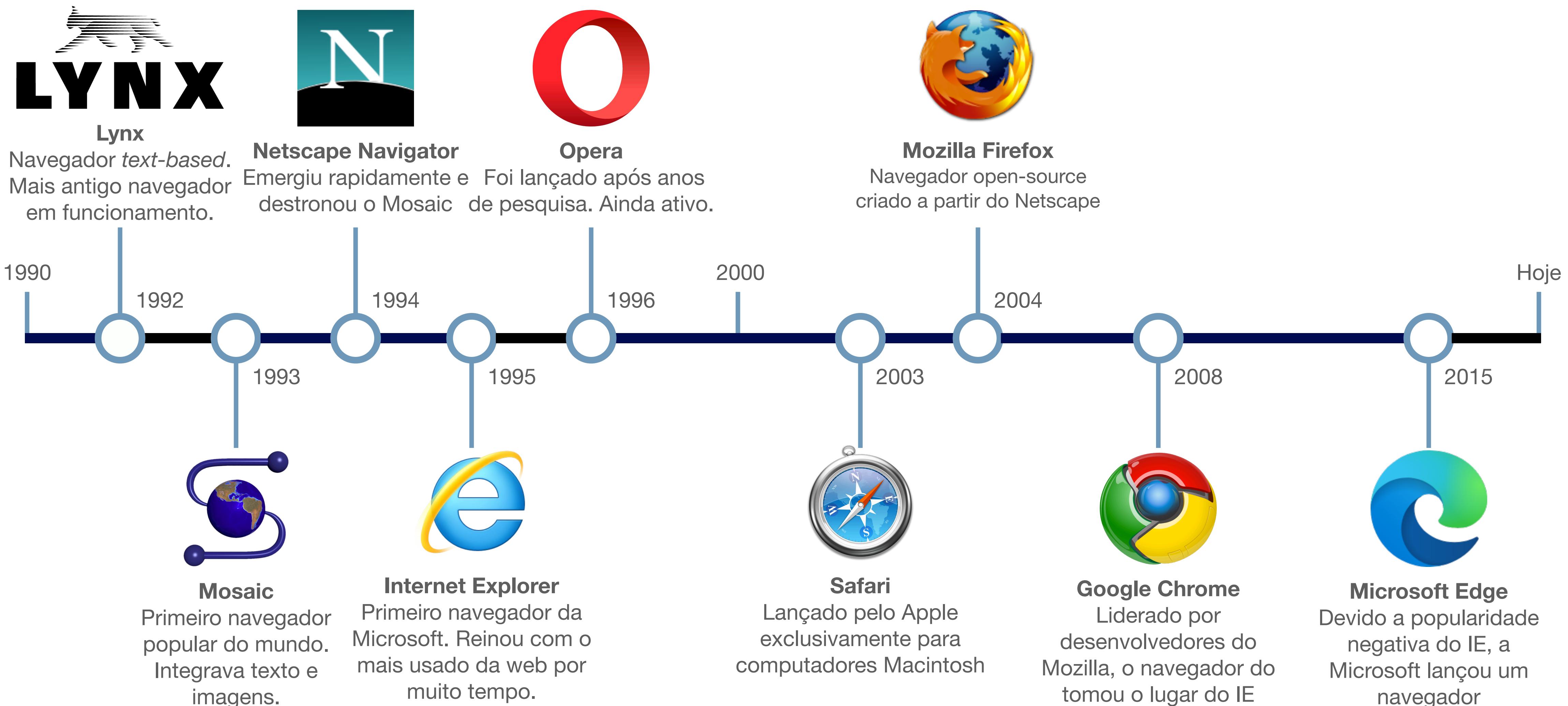
Interpretador de JavaScript

UI Backend

Usado para desenhar componentes gráficos básicos. Utiliza a API do S.O

A World Wide Web

Navegadores



A World Wide Web

Servidores web

- É um programa que fica aguardando requisições por recursos
 - HTML, arquivos de áudio, imagens ...
- Tais requisições são feitas utilizando protocolo HTTP
- Tem domínio sobre a base de informações mantidas naquele sítio
- Em geral, implementa algum tipo de controle de acesso a documentos

A World Wide Web

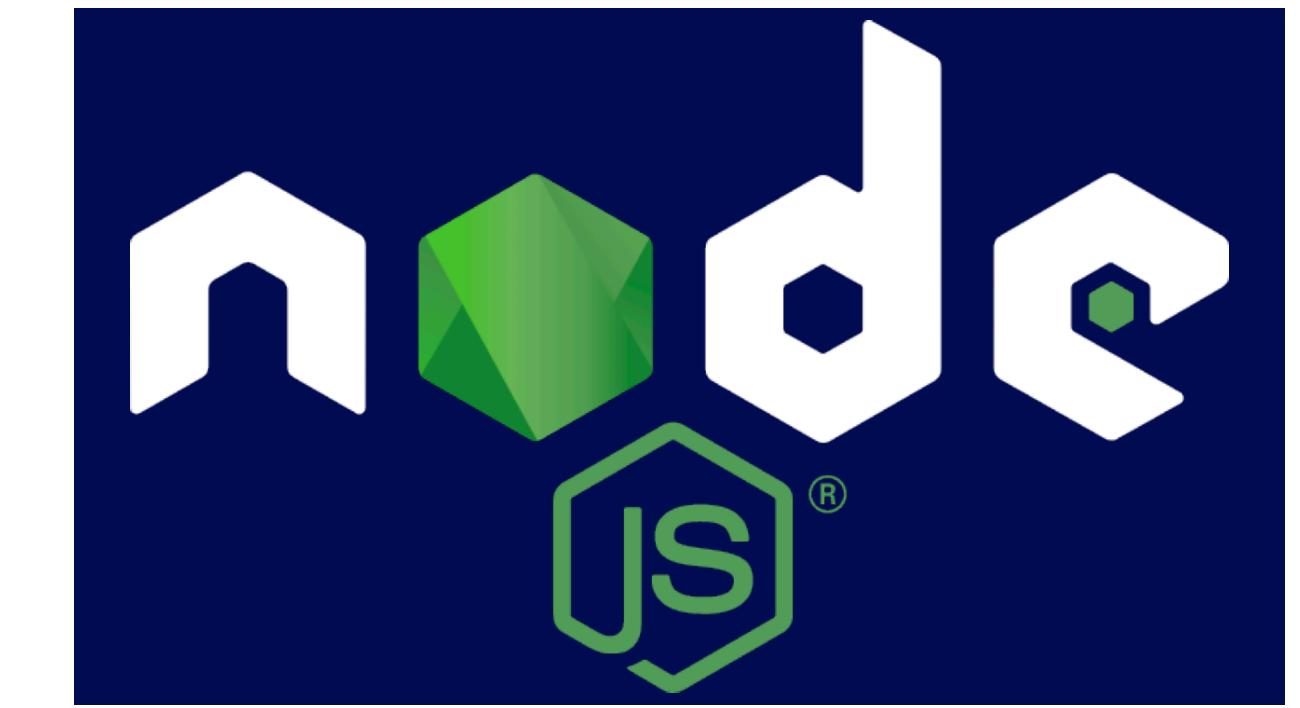
Servidores web



Logotipo do Apacher Server



Logotipo do Nginx Server



Logotipo do NodeJs



Logotipo do Tomcat Server



OpenLiteSpeed

Logotipo do OpenLiteSpeed



Logotipo do Lighttpd Server

A World Wide Web

Uniform Resource Identifier

- Identificador Uniforme de Recurso
 - É uma cadeia de caracteres usada para **identificar** ou **denominar** um recurso na Internet
- Pode ser classificada como um nome (**URN**) ou um localizador (**URL**), ou ainda como ambos
- Exemplos:
 - URN
 - urn:isbn:0-486-27557-4
 - urn:issn:1535-3613
 - URL
 - <http://example.org/absolute/path/to/resource.txt>
 - <ssh://bruno@example.org:3000>

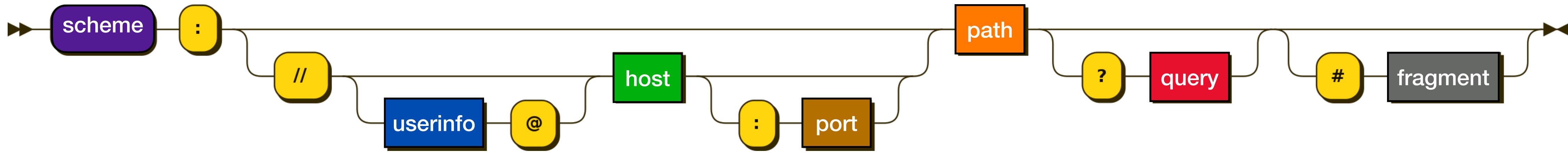
A World Wide Web

Uniform Resource Locator

- Localizador de Recurso Universal
 - É o endereço de um recurso disponível em uma rede
 - Identifica um recurso, e prover meios de agir sobre, obter e representar o recurso
 - Especifica a localização na "rede".

A World Wide Web

Uniform Resource Locator



http://brunomateus@www.meusite.com.br:123/aulas/disciplinas?nome=web#01

Esquema Nome do usuário Nome do host porta *path* *query* *fragmento*

http://www.meusite.com.br/aulas/disciplinas/web.html

Esquema Nome do host *path*

A World Wide Web

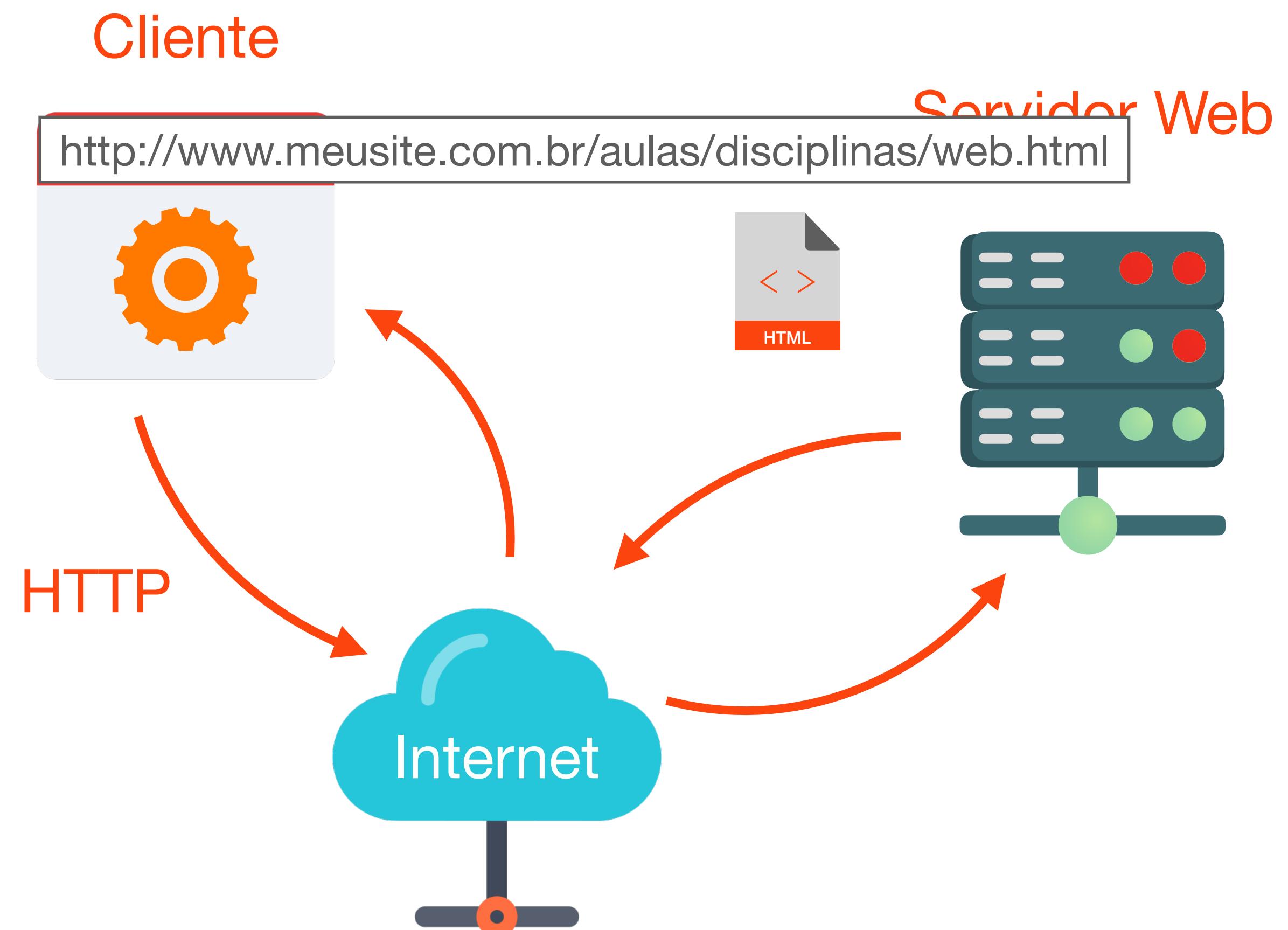
Domain Name System (DNS)

- Sistema hierárquico e distribuído de gestão de nomes
- Mapeia os nomes de domínios para endereços IP
 - `www.meusite.com.br.` -> `200.17.41.185`
- A maioria dos sistemas operacionais mantém um cache local chamados de hosts file
 - Windows: `C:/Windows/system32/drivers/etc/hosts`
 - Mac: `/private/etc/hosts`
 - Linux: `/etc/hosts`

A World Wide Web

- Cenário mais comum:

1. O navegador digita a **URL** de um recurso na barra de endereço o navegador requisita um recurso ao servidor
 - 1.1.O sistema operacional usando **DNS** descobre o endereço **IP** do domínio em que o recurso está localizado
 - 1.2.Uma requisição **HTTP** é feita via **GET** em busca do recurso **web.html** localizado em **/aulas/disciplinas**
2. O servidor web interpreta a requisição
3. O servidor gera uma resposta **HTTP** para o cliente
4. O envia a resposta ao cliente
5. O navegador processa a resposta **HTTP** recebida
6. O navegador interpreta o conteúdo **HTML**
7. O navegador renderiza o conteúdo na tela apresentando-o ao usuário



O Protocolo HTTP

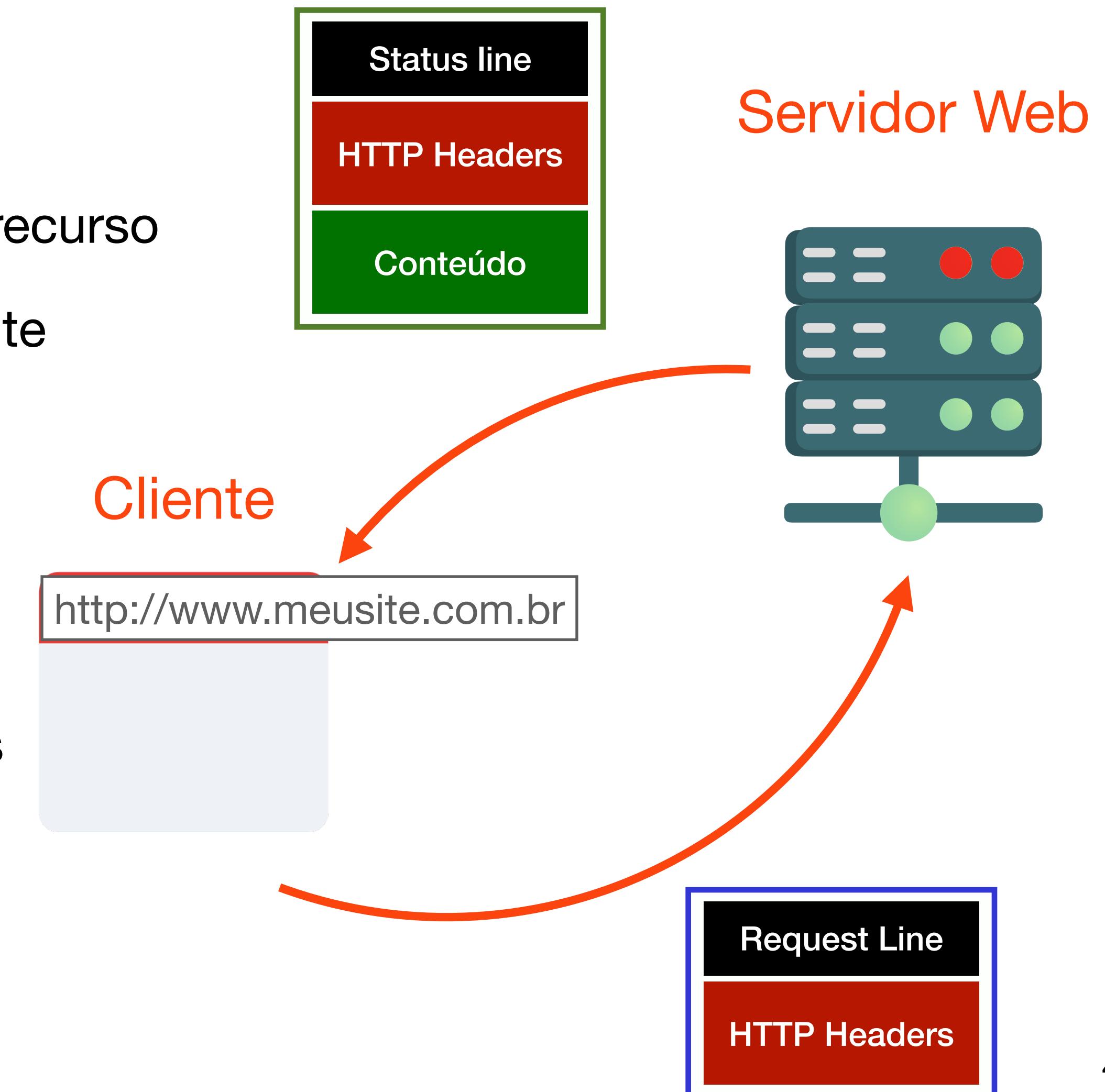


O protocolo HTTP

- Hypertext Transfer Protocol (Protocolo de Transferência de Hipertexto)
- Usado desde 1990 é protocolo de comunicação utilizado para transferir dados por intranets e World Wide Web
 - Normalmente o servidor usa a porta 80
- Responsável pelo tratamento de requisições / respostas entre cliente e servidor Web
- Serve para transferência de texto e dados binários

O protocolo HTTP

- Funcionamento:
 - O cliente envia uma mensagem de requisição de um recurso
 - O servidor envia uma mensagem de resposta ao cliente
- Composição das mensagens:
 - Cabeçalho e Corpo (opcional em certos casos)
 - Stateless(Sem estados)
 - Não existe informações sobre o estado das conexões
 - Próxima ação independe da anterior
 - Realiza apenas uma ação por vez



O protocolo HTTP

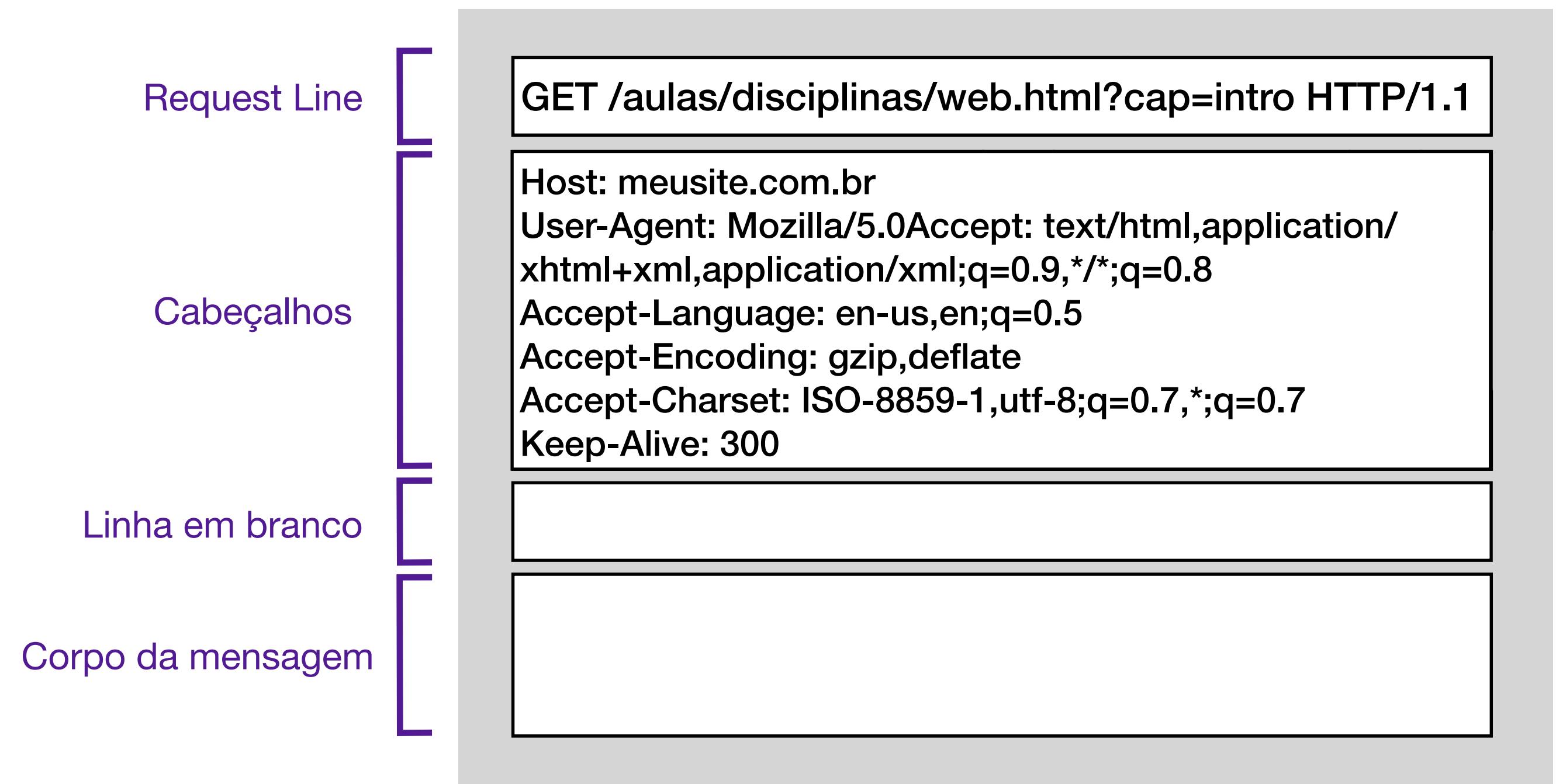
Métodos

- **GET** – Solicita algum recurso
 - Dados são anexados à URL, ficando visíveis ao usuário
 - **POST** – Envia dados referentes ao recurso especificado para serem processados
 - Dados são incluídos no corpo do comando
 - **PUT** – Envia certo recurso
 - **DELETE** – Exclui o recurso
-
- **HEAD** – Variação do GET em que o recurso não é retornado
 - **TRACE** – Ecoa o pedido, de maneira que o cliente possa saber o que os servidores intermediários estão mudando em seu pedido
 - **OPTIONS** – Recupera os métodos HTTP que o servidor aceita
 - **CONNECT** – Converte a conexão de requisição em um túnel TCP/IP transparente, geralmente para facilitar a comunicação encriptada através de um proxy HTTP sem encriptação
 - **PATCH** – Usado para aplicar modificações parciais em um recurso

O protocolo HTTP

Requisição

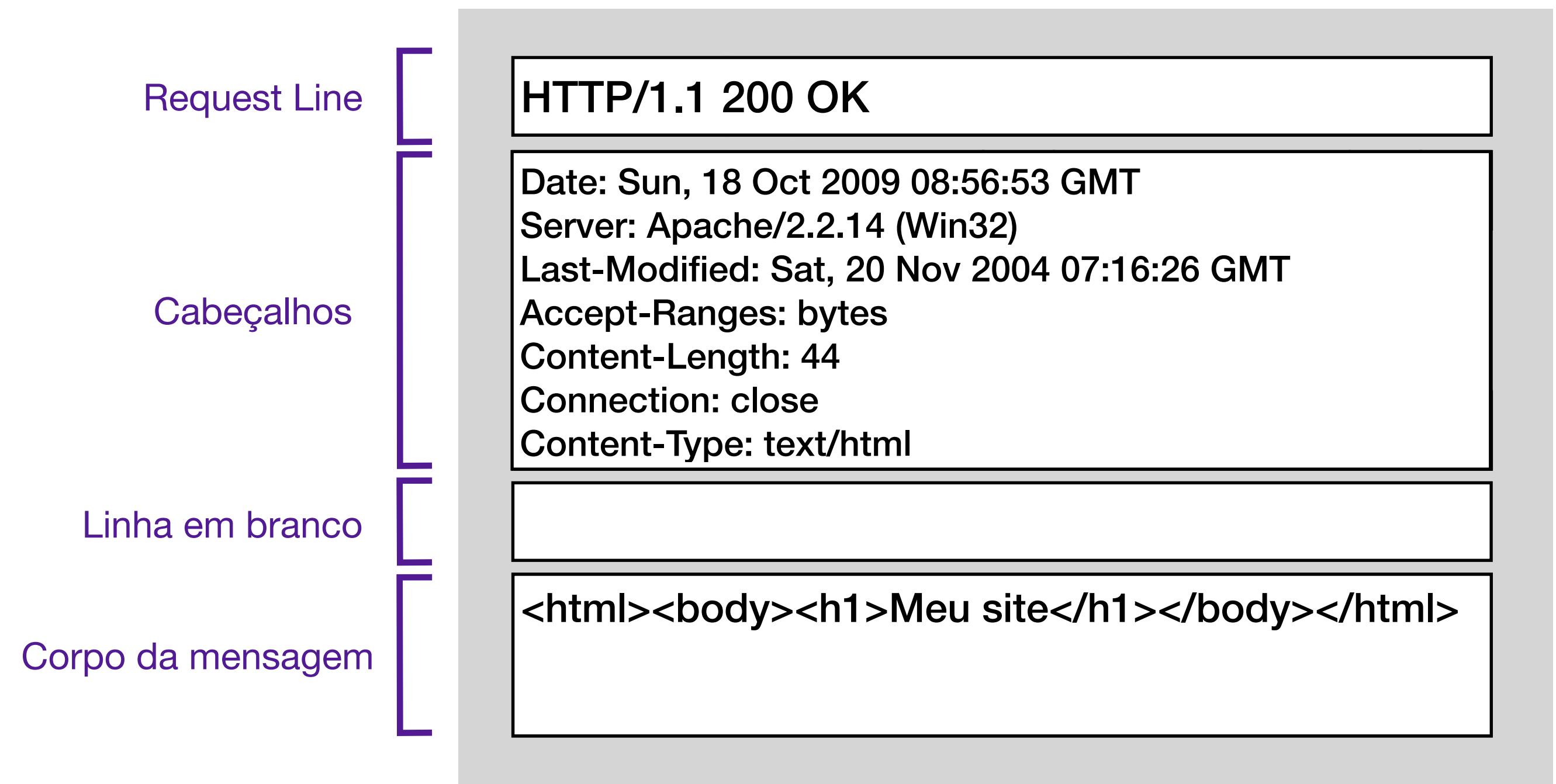
- Linha inicial
 - Método
 - URL
 - Versão do HTTP utilizado
- Linhas de cabeçalhos
- Linha em branco obrigatória
- Corpo de mensagem opcional



O protocolo HTTP

Resposta

- Linha inicial ou linha de status
 - Versão do protocolo HTTP
 - Código de status da resposta
 - Frase relacionada com status
- Linhas de cabeçalhos
- Linha em branco obrigatória
- Corpo de mensagem opcional



O protocolo HTTP

Código de Status

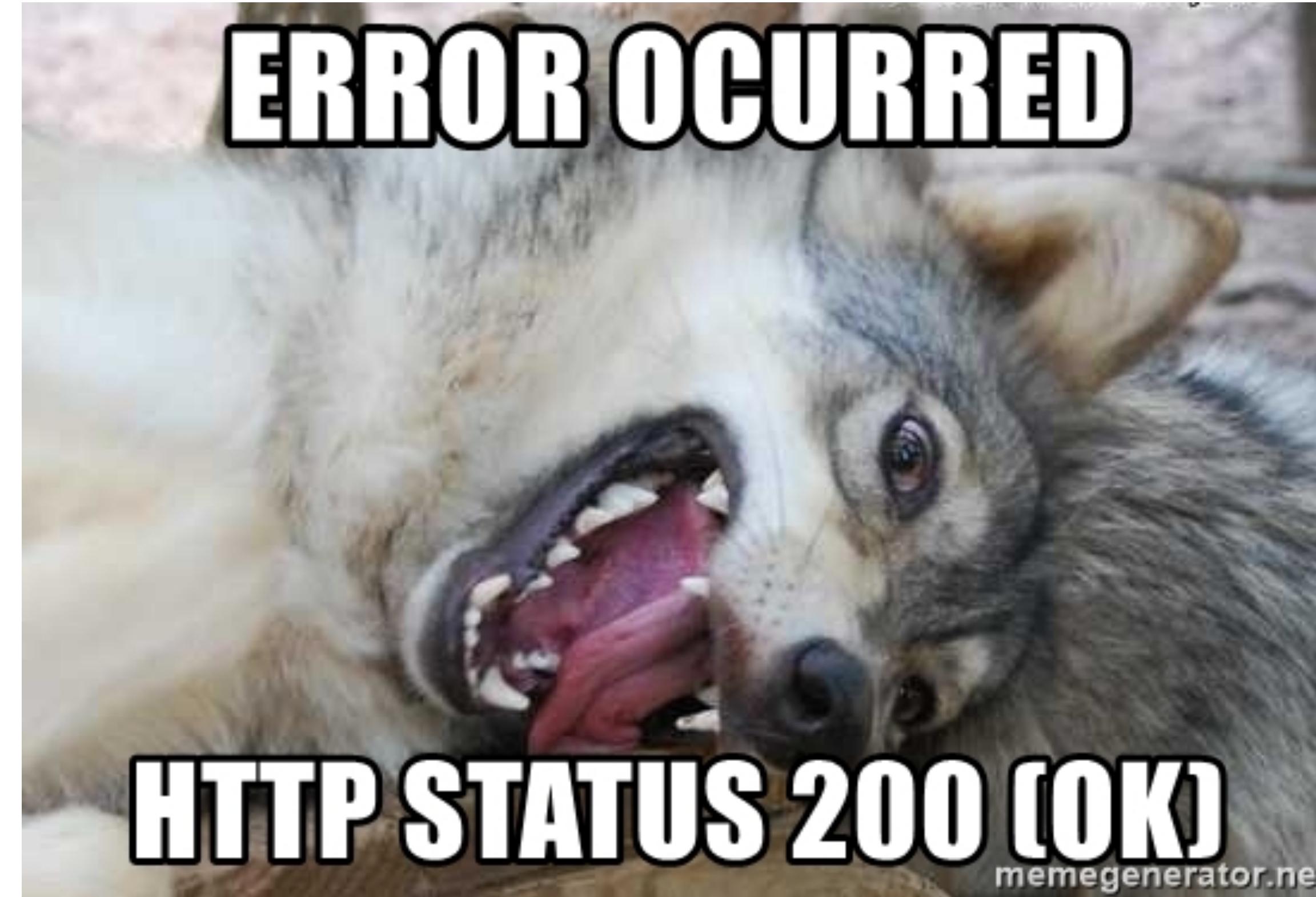
- É formado por três dígitos e o primeiro dígito representa a classe. São classificadas em cinco classes:

Classe/Código	Utilização
1xx: Informational (Informação)	Utilizada para enviar informações para o cliente de que sua requisição foi recebida e está sendo processada
2xx: Success (Sucesso)	Indica que a requisição do cliente foi bem sucedida
3xx: Redirection (Redirecionamento)	Informa a ação adicional que deve ser tomada para completar a requisição
4xx: Client Error (Erro no cliente)	Avisa que o cliente fez uma requisição que não pode ser atendida
5xx: Server Error (Erro no servidor)	Ocorreu um erro no servidor ao cumprir uma requisição válida

- O protocolo HTTP define somente alguns códigos em cada classe, mas cada servidor pode definir seus próprios códigos

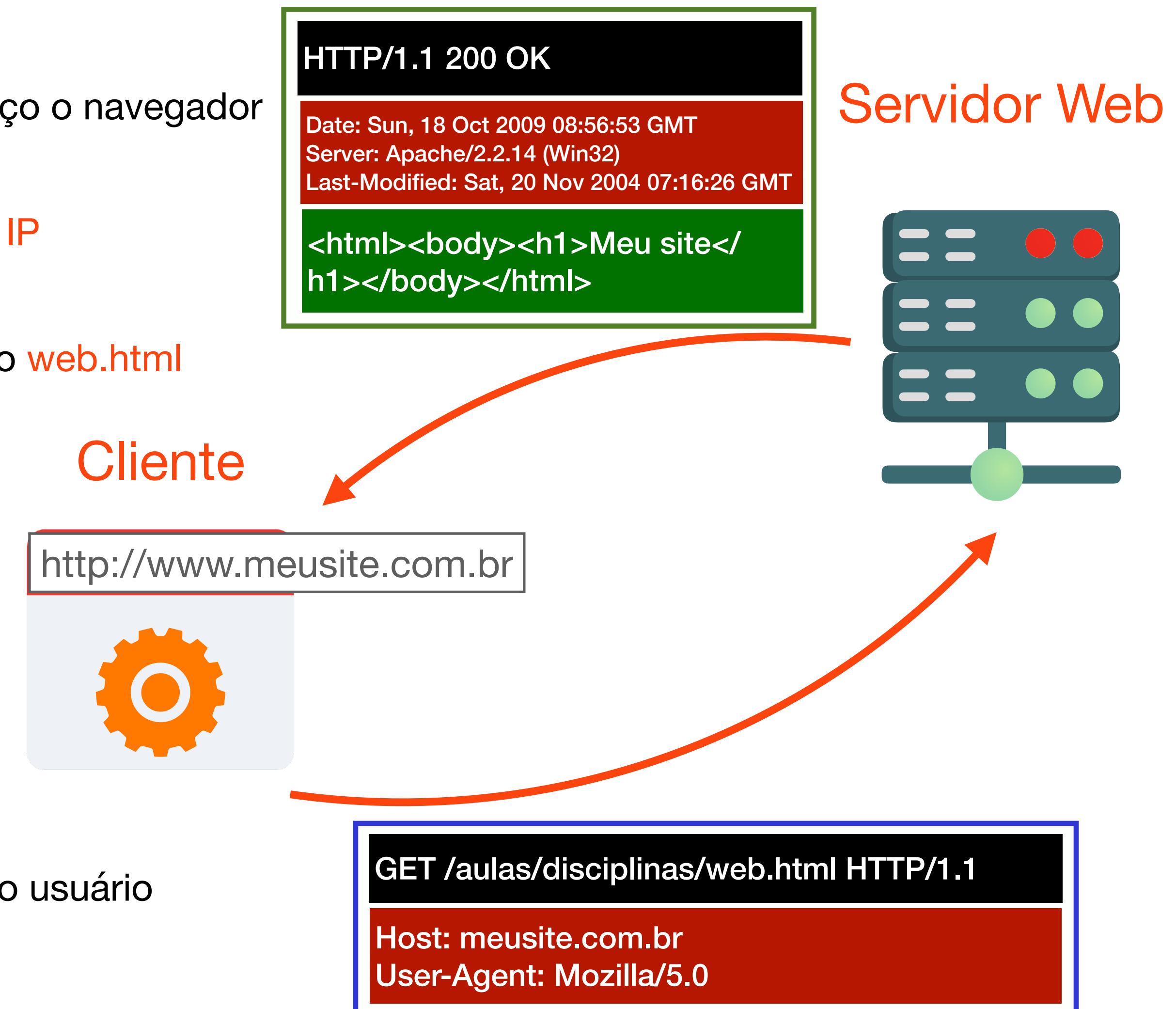
O protocolo HTTP

Código de Status



O protocolo HTTP

1. O navegador digita a **URL** de um recurso na barra de endereço o navegador requisita um recurso ao servidor
 - 1.1.O sistema operacional usando **DNS** descobre o endereço **IP** do domínio em que o recurso está localizado
 - 1.2.Uma requisição **HTTP** é feita via **GET** em busca do recurso **web.html** localizado em **/aulas/disciplinas**
2. O servidor web interpreta a requisição
3. O servidor gera uma resposta **HTTP** para o cliente
4. O envia a resposta ao cliente
5. O navegador processa a resposta **HTTP** recebida
6. O navegador interpreta o conteúdo **HTML**
7. O navegador renderiza o conteúdo na tela apresentando-o ao usuário



Referências

- Quem inventou a internet? Conheça a sua história de origem
- A história da Internet: pré-década de 60 até anos 80
- O Modelo Cliente Servidor
- Client-Server Architecture
- What is Client-Server Architecture? Everything You Should Know
- A Timeline of the Rise and Fall of Popular Web Browsers

Por hoje é só