



# Comunicação na Internet

**Sistemas Distribuídos e Mobile**

Prof. Gustavo Torres Custodio  
gustavo.custodio@anhembi.br

# Agenda

- Redes em Sistemas Distribuídos
- Internet
- Protocolos
  - OSI
- World Wide Web
- HTTP



# Redes em Sistemas Distribuídos

- As redes usadas em sistemas distribuídos incluem:
  - fios, cabos, fibra e canais sem fio;
  - dispositivos de hardware, incluindo roteadores, switches, pontes, hubs, repetidores e interfaces de rede; e
  - componentes de software, incluindo pilhas de protocolo, manipuladores de comunicação e drivers.
- A funcionalidade e o desempenho resultantes disponíveis para sistemas distribuídos e programas aplicativos são afetados por tudo isso.

# Definições – Redes em Sistemas Distribuídos

- Subsistema de comunicação:
  - Coleção de componentes de hardware e software que fornecem os recursos de comunicação para um sistema distribuído
- Hosts:
  - Os computadores e outros dispositivos que usam a rede para fins de comunicação.
- Nó:
  - Usado para se referir a qualquer computador ou dispositivo de comutação conectado a uma rede.

# Definições – Redes em Sistemas Distribuídos

- Transmissão de pacotes

- Transmissão de unidades lógicas de informação, ou mensagens
  - sequências de itens de dados de comprimento arbitrário.
- Antes de uma mensagem ser transmitida, ela é subdividida em pacotes.
- Pacote (mais simples)
  - sequência de dados binários (uma matriz de bits ou bytes) de comprimento restrito, e
  - informações de endereçamento suficientes para identificar os computadores de origem e destino.





- A Internet é um único subsistema de comunicação que fornece comunicação entre todos os hosts que estão conectados a ela.
- A Internet é construída a partir de muitas sub-redes.
  - Uma sub-rede é uma unidade de roteamento (entrega de dados de uma parte da Internet para outra); é uma coleção de nós que podem ser alcançados na mesma rede física.
- A infraestrutura da Internet inclui uma arquitetura e componentes de hardware e software que integram efetivamente diversas sub-redes em um único serviço de comunicação de dados.
- O projeto de um subsistema de comunicação é fortemente influenciado pelas características dos sistemas operacionais utilizados nos computadores que compõem o sistema distribuído, bem como pelas redes que os interligam.

- Internetworking

- É um subsistema de comunicação no qual várias redes estão conectadas para fornecer recursos comuns de comunicação de dados que sobrepõem as tecnologias e protocolos das redes de componentes individuais e os métodos usados para sua interconexão.

- Por que usá-la em Sistemas Distribuídos?

- São necessárias para o desenvolvimento de sistemas distribuídos abertos e extensíveis.

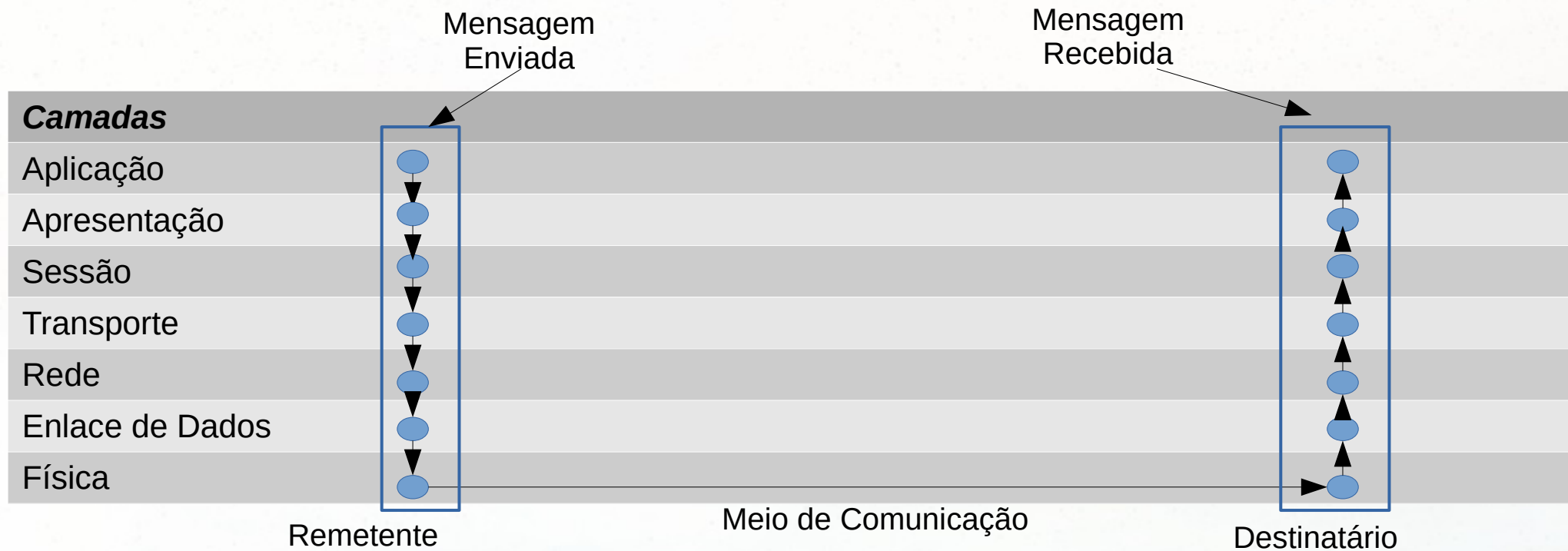
# Protocolos: Conceitos Básicos

- Conjunto de regras, procedimentos e formatos para garantir a comunicação entre duas entidades geograficamente distintas
  - A sequência de mensagens que devem ser trocadas;
  - O formato dos dados nas mensagens.
- ISO (International Standard Organization), desenvolveu um modelo de referência:
  - diversos níveis envolvidos;
  - o que deve ser tratado em cada nível;
  - Open Systems Interconnection Reference Model, ISO OSI ou, simplesmente, Modelo OSI.



# Protocolos: Camadas de Comunicação

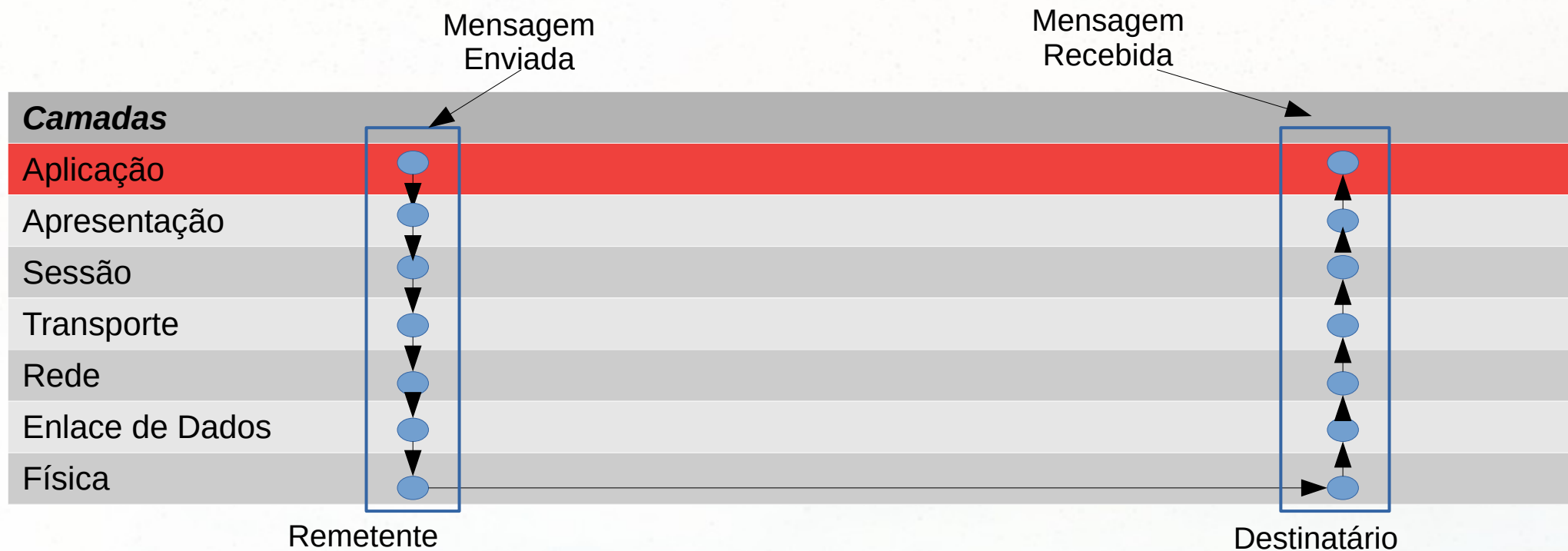
- Comunicação é dividida em sete níveis ou camadas, onde cada camada trata de um aspecto específico da comunicação.



# Protocolos: Camadas de Comunicação

- Aplicação:

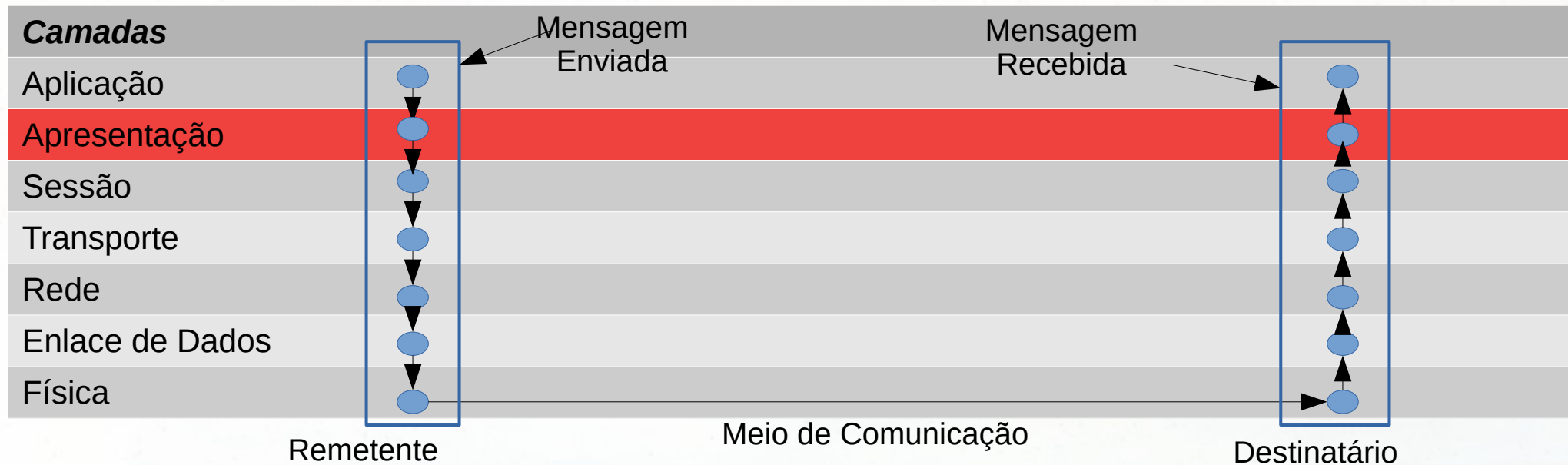
- Esta camada é a porta de entrada para a rede ou o sistema de comunicação, da forma como é vista pelos aplicativos que usam este sistema, ou seja, fornece um conjunto de funções para serem usadas pelos aplicativos.



# Protocolos: Camadas de Comunicação

- Apresentação

- A camada de apresentação garante que a informação seja divulgada pela camada de aplicação legivelmente para outro sistema, sendo assim, codifica e converte dados com o propósito de fazer com que os sistemas falem a mesma língua.
- Suas principais funcionalidades são:
  - formatação de dados; criptografia de dados; compactação de dados.

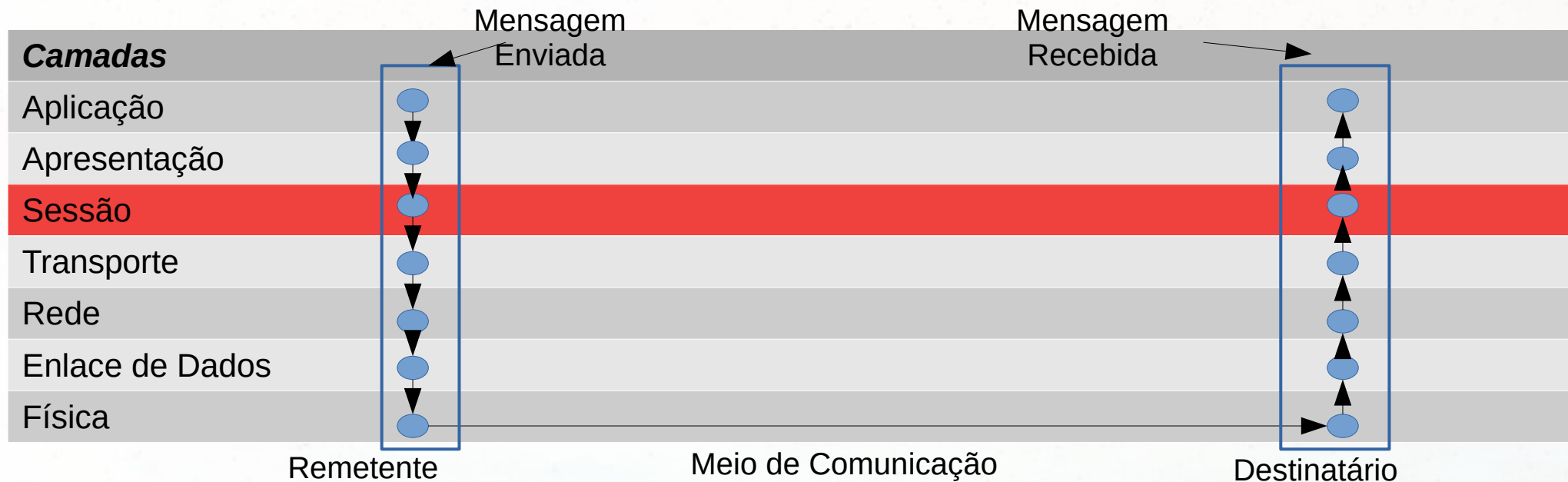




# Protocolos: Camadas de Comunicação

- Sessão:

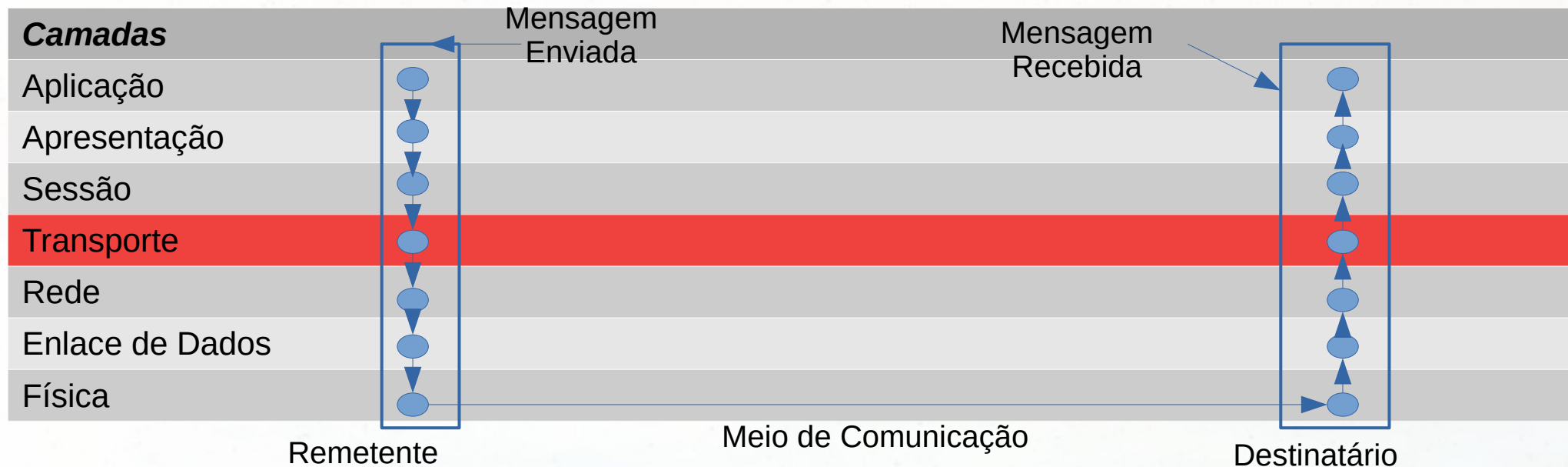
- é responsável pela inicialização, gerenciamento e finalização de sessões entre o transmissor e receptor.
- Suas funcionalidades são:
  - fornecer seus serviços à camada de apresentação, manter os dados de diferentes aplicações separados uns dos outros, ou seja, oferecer recursos e serviços eficientes nos diálogos e conversações sobre a camada de sessão.



# Protocolos: Camadas de Comunicação

- Transporte:

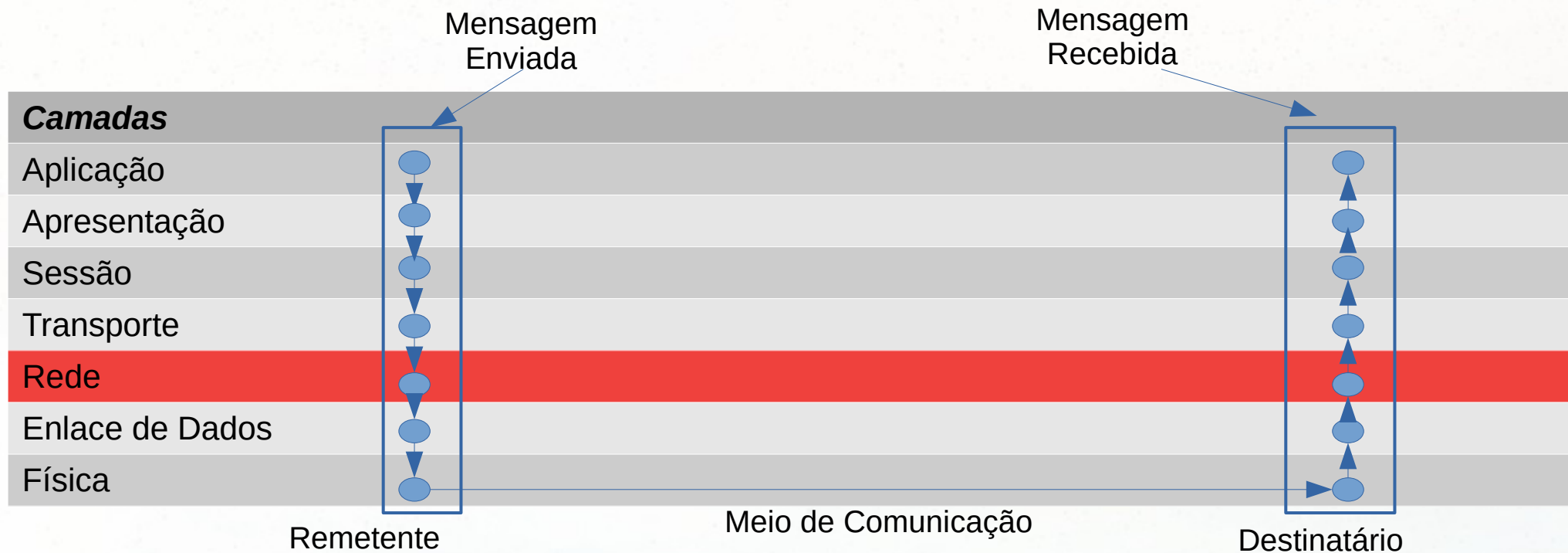
- segmenta os dados e reconstrói os fluxos de dados provenientes de camadas superiores. Também provém de comunicação ponto a ponto, onde estabelece uma conexão lógica entre aplicações origem e destino em uma rede;
- estabelece, mantém e termina corretamente os circuitos virtuais, e controle de fluxo de informação, detecção e recuperação de erros de transporte, garantindo qualidade nos serviços e confiabilidade.



# Protocolos: Camadas de Comunicação

- Rede:

- é responsável pelo encaminhamento dos dados através da interligação de redes, endereçamento de pacotes de dados, e conversão de endereços lógicos (IP) em endereços físicos ou MAC.

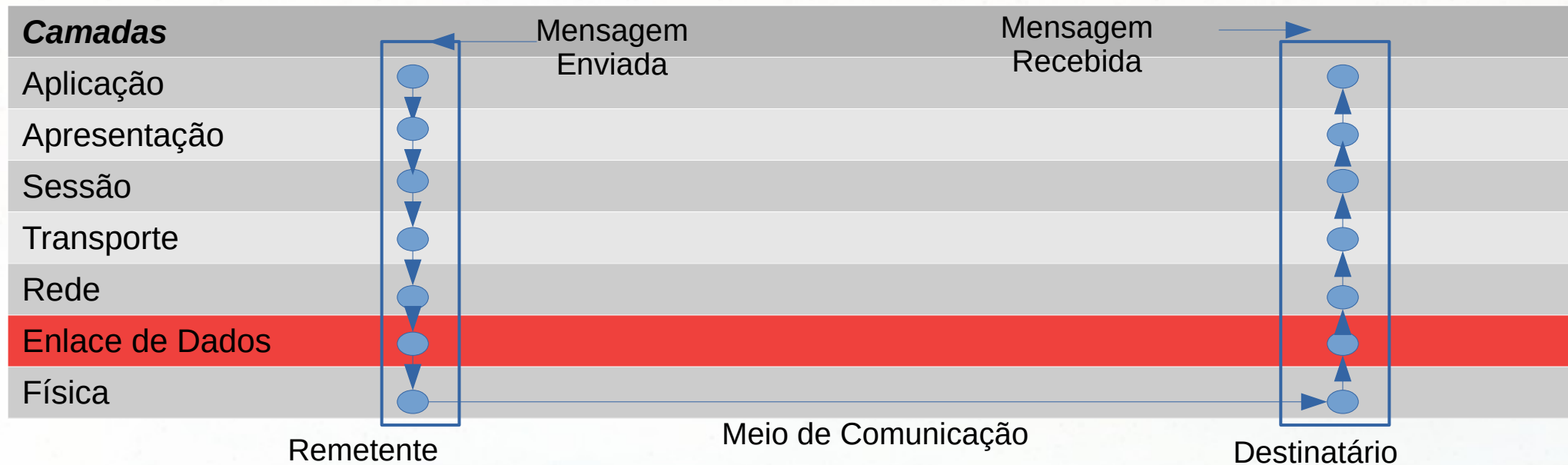




# Protocolos: Camadas de Comunicação

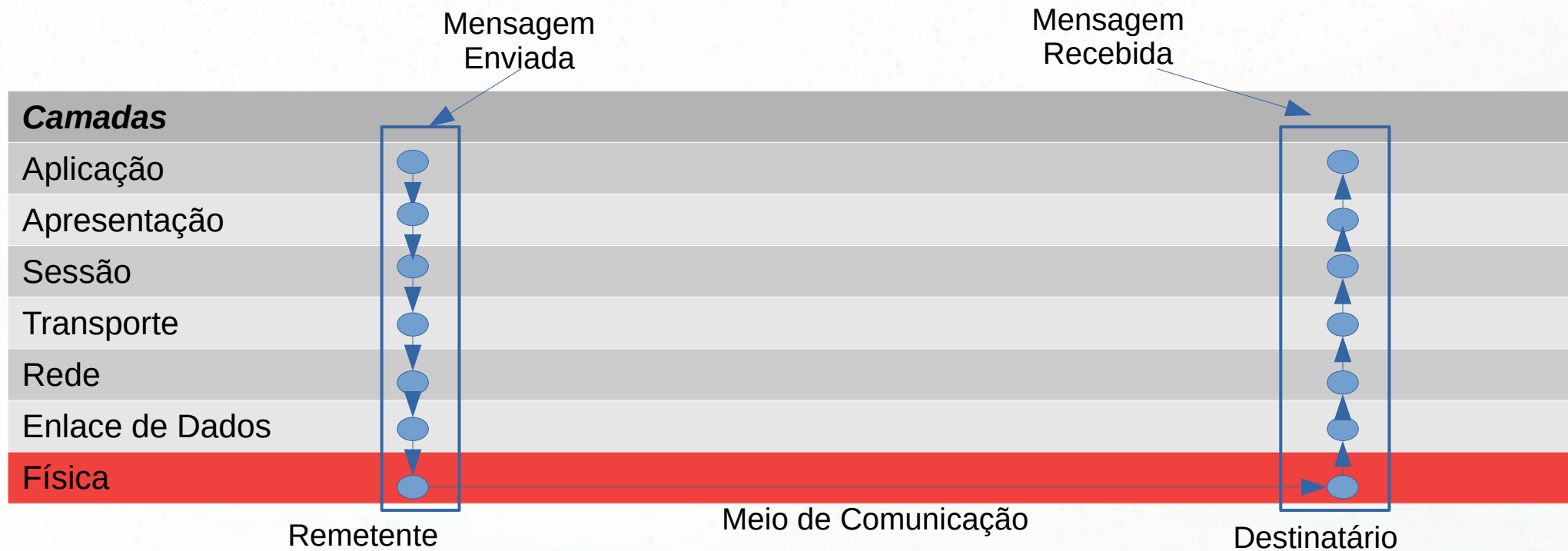
- Enlace:

- oferece aos dados segurança, conversão em bits dos pacotes vindos da camada de rede, realizando em seguida a transmissão através de um link físico (cabeamento).
- Os serviços prestados pela camada de enlace são dependentes dos protocolos que provêm entrega garantida entre enlaces, ou seja, desde o transmissor, passando por um único enlace, até chegar ao receptor.



# Protocolos: Camadas de Comunicação

- Física:
  - Nesta camada são definidas especificações elétricas, mecânicas, funcionais e de procedimentos.



# World Wide Web

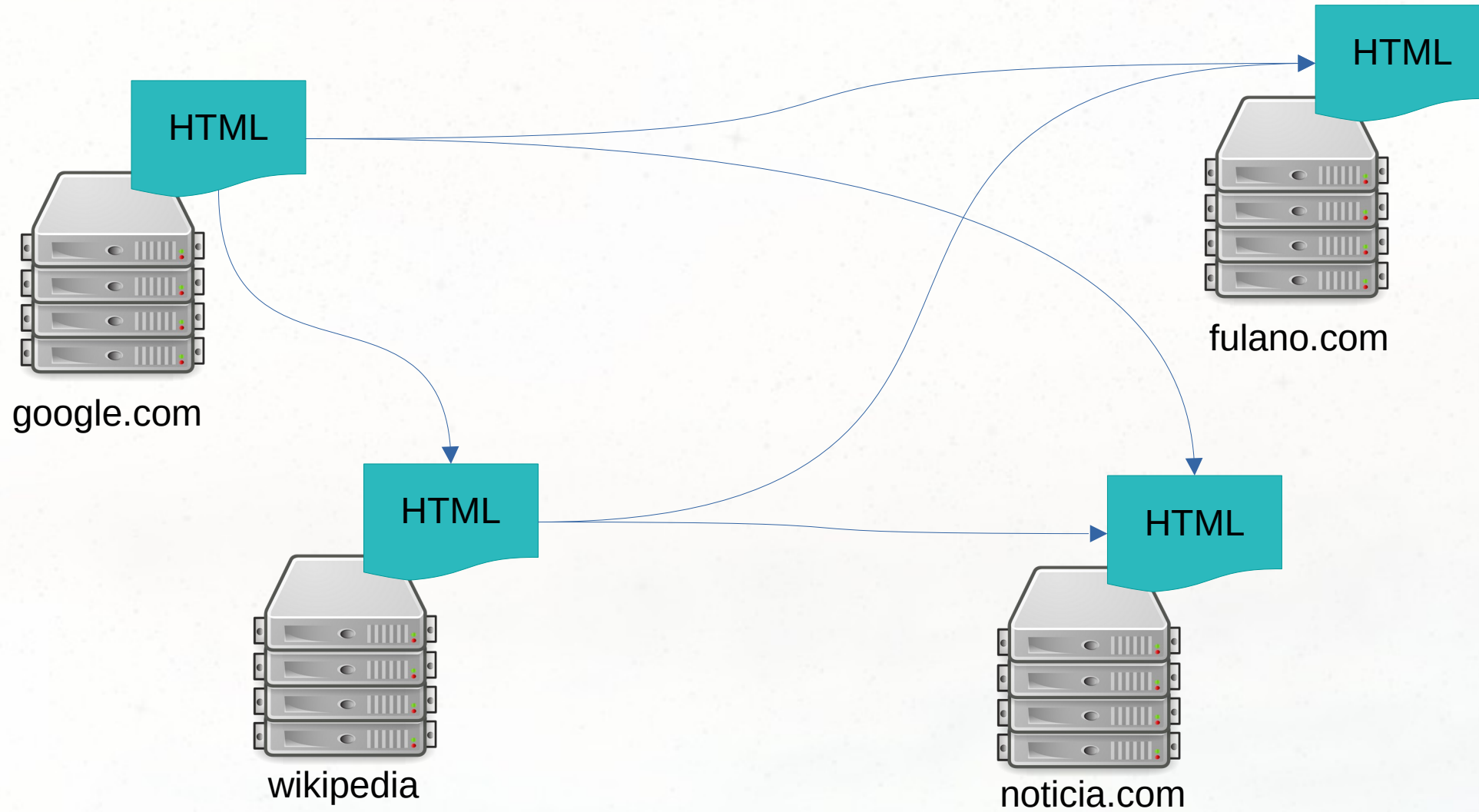
- Criado por Tim Berners-Lee e Robert Cailliau
  - Em 1989
  - CERN (European Organization for Nuclear Research)
- Ideia original:
  - Compartilhar artigos científicos
- Objetivo:
  - Distribuir documentos na rede;
  - Interligar recursos e documentos;
  - Formato universal para documentos.



Robert Cailliau, Jean-François Abramatic e Tim Berners-Lee.

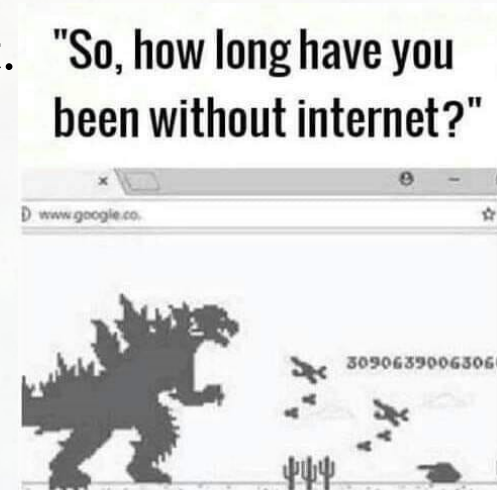


# World Wide Web



# Internet e WWW

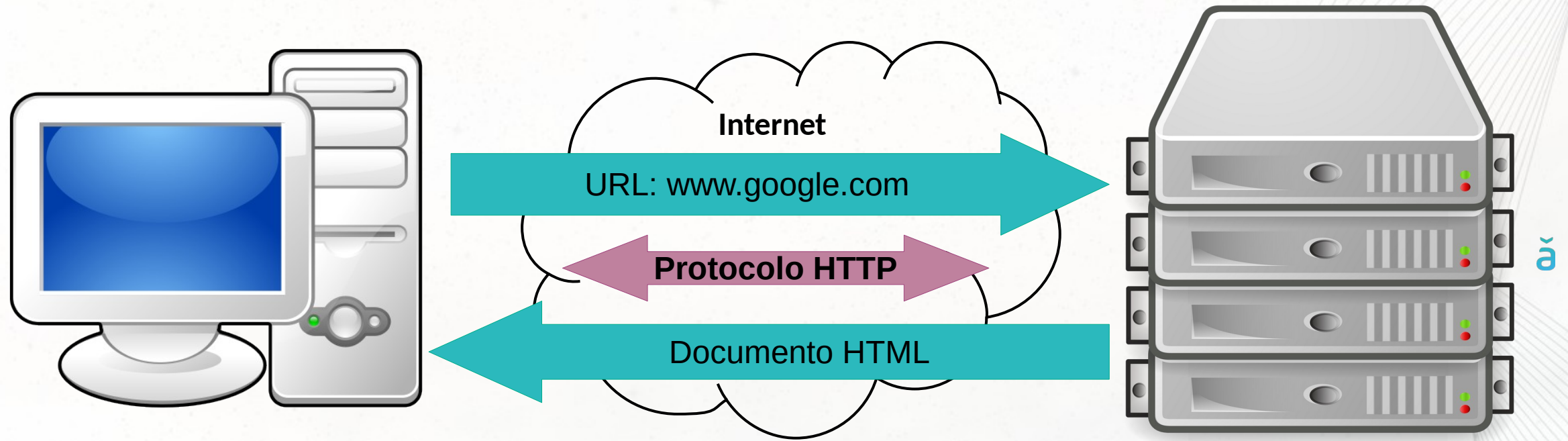
- Qual a diferença entre Internet e WWW (World Wide Web)??
  - Internet:
    - A Internet fornece a interligação dos computadores. Transporta informação.
    - A Internet transporta os documentos que existem na World Wide Web.
  - WWW:
    - A World Wide Web é um sistema de documentos hipertexto interligados.
    - A World Wide Web é um dos vários serviços que funciona sobre a Internet.



- TCP/IP: Transmission Control Protocol/Internet Protocol
  - Infraestrutura de rede
- Arquitetura Cliente/Servidor:
  - Estratégia para implementação dos serviços
- HTTP: Hypertext Transfer Protocol
  - Protocolo da camada de aplicação
- HTML: Hypertext Markup Language
  - Padrão de apresentação dos dados
- MIME: Multipurpose Internet Mail Extensions
  - Padrão de codificação dos dados



# Arquitectura World Wide Web



- Hyper Text Markup Language

- Linguagem de marcação padrão para criar páginas da Web
- Descreve a estrutura de uma página da Web
- Consiste em uma série de elementos
- Elementos HTML informam ao navegador como exibir o conteúdo
- Elementos HTML rotulam partes do conteúdo como "este é um título", "este é um parágrafo", "este é um link", etc.

```
<!DOCTYPE html>
<html>
<head>
<title>Page Title</title>
</head>
<body>

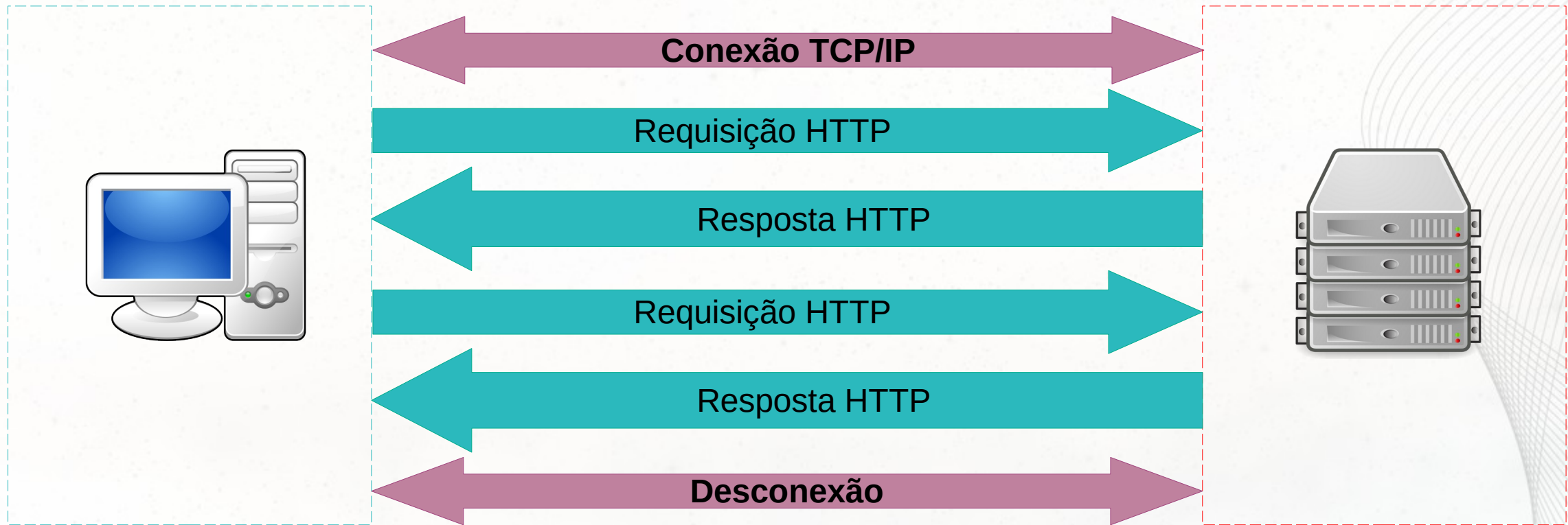
<h1>My First Heading</h1>
<p>My first paragraph.</p>

</body>
</html>
```

- Hypertext Transfer Protocol
- Os documentos são transferidos entre o servidor web e o navegador web.
- A forma como se processa esta transferência é definida pelo protocolo HTTP, pertencente à família TCP/IP.



# HTTP



- Tipos de mídia
  - (também conhecido como Multipurpose Internet Mail Extensions ou tipo de MIME)
- Indica a natureza e o formato de um documento, arquivo ou variedade de bytes.
- Os tipos de MIME são definidos e padronizados na RFC 6838 da IETF.
- Exemplos:
  - text/plain
  - text/css
  - text/html
  - text/javascript
- Exemplos:
  - image/apng
  - image/avif
  - image/gif
  - image/jpeg
  - image/png
- Exemplos:
  - audio/wave
  - audio/wav
  - audio/x-wav
  - audio/x-pn-wav
  - video/webm
  - video/ogg

- JSON (JavaScript Object Notation) é um formato leve de troca de dados.
- É fácil para humanos ler e escrever.
- É fácil para as máquinas analisar e gerar.
- Ele é baseado em um subconjunto do JavaScript Programming Language Standard
  - Dezembro de 1999





- JSON é um formato de texto completamente independente de linguagem,
  - mas usa convenções familiares aos programadores da família das linguagens do C:
    - C, C++, C#, Java, JavaScript, Perl, Python e muitos outros.
- Essas propriedades tornam o JSON uma linguagem de troca de dados ideal.



- JSON é construído em duas estruturas:
  - Uma coleção de pares nome/valor.
    - Objeto, registro, estrutura, dicionário, tabela de hash, lista de chaves ou matriz associativa.
  - Uma lista ordenada de valores.
    - Matriz, vetor, lista ou sequência.
- Estas são estruturas de dados universais.
  - Praticamente todas as linguagens de programação modernas as suportam de uma forma ou de outra.
  - Faz sentido que um formato de dados intercambiável com linguagens de programação também seja baseado nessas estruturas.



# JSON x XML

```
{
  "glossary": {
    "title": "example glossary",
    "GlossDiv": {
      "title": "S",
      "GlossList": {
        "GlossEntry": {
          "ID": "SGML",
          "SortAs": "SGML",
          "GlossTerm": "Standard Generalized Markup Language",
          "Acronym": "SGML",
          "Abbrev": "ISO 8879:1986",
          "GlossDef": {
            "para": "A meta-markup language, used to create markup languages such as DocBook.",
            "GlossSeeAlso": ["GML", "XML"]
          },
          "GlossSee": "markup"
        }
      }
    }
  }
}
```

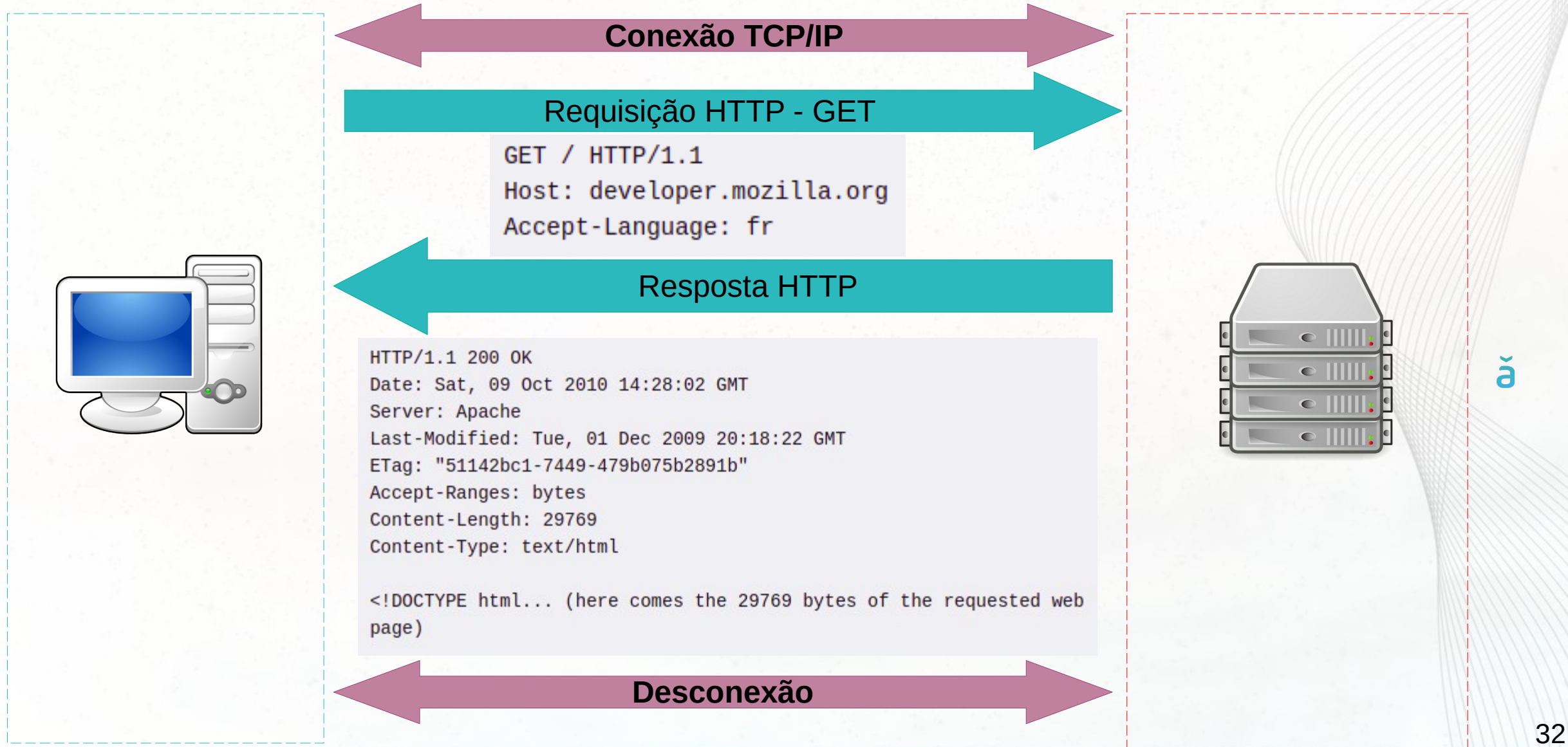


# JSON x XML

```
<!DOCTYPE glossary PUBLIC "-//OASIS//DTD DocBook V3.1//EN">
<glossary><title>example glossary</title>
  <GlossDiv><title>S</title>
    <GlossList>
      <GlossEntry ID="SGML" SortAs="SGML">
        <GlossTerm>Standard Generalized Markup Language</GlossTerm>
        <Acronym>SGML</Acronym>
        <Abbrev>ISO 8879:1986</Abbrev>
        <GlossDef>
          <para>A meta-markup language, used to create markup
languages such as DocBook.</para>
          <GlossSeeAlso OtherTerm="GML">
            <GlossSeeAlso OtherTerm="XML">
              </GlossDef>
            <GlossSee OtherTerm="markup">
              </GlossEntry>
            </GlossList>
          </GlossDiv>
        </glossary>
```

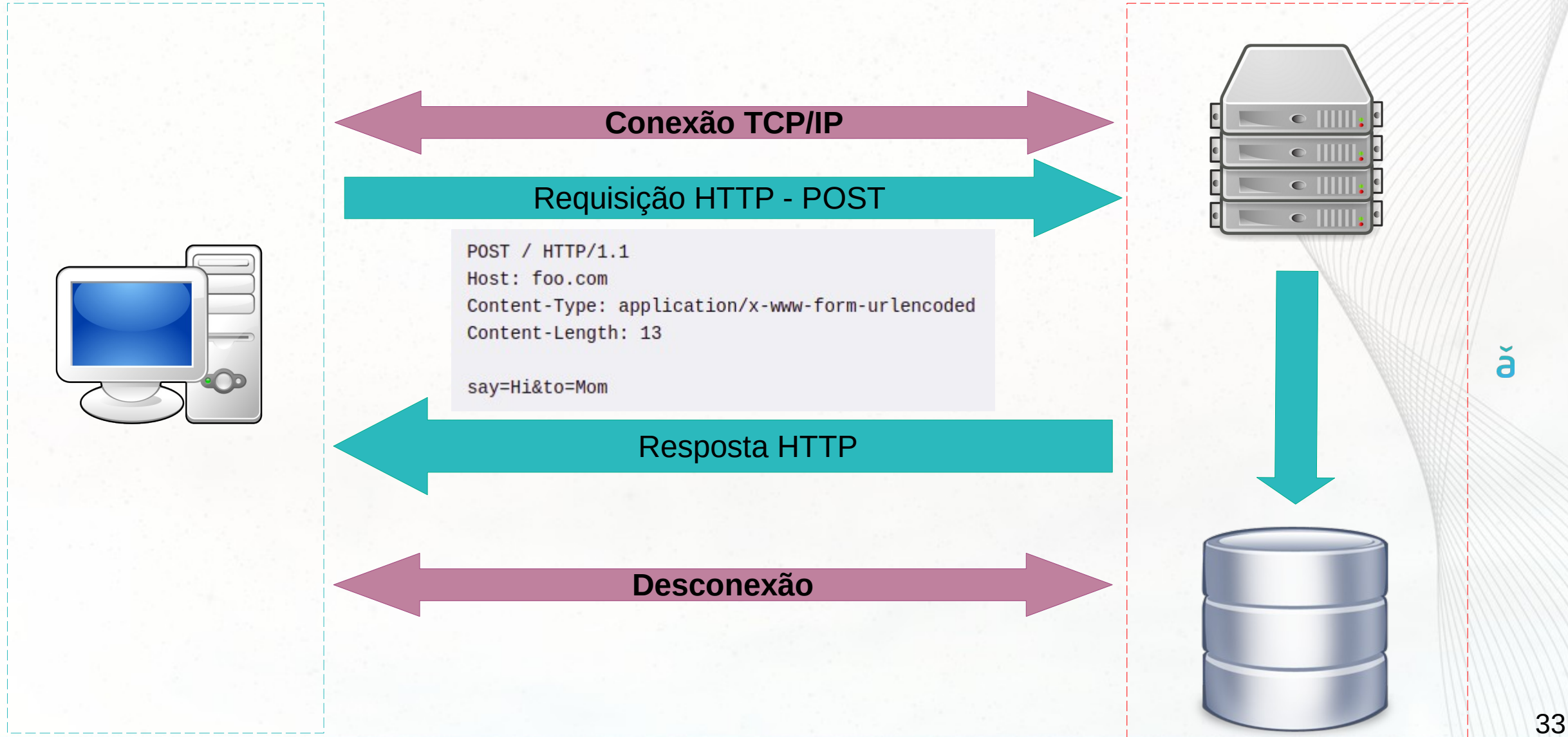
- Os principais serviços do HTTP incluem:
  - GET: solicita ao servidor o envio de um recurso; é o serviço essencial para o protocolo;
  - HEAD: variante de GET que solicita ao servidor o envio apenas de informações sobre o recurso;
  - PUT: permite que o cliente autorizado armazene ou altere o conteúdo de um recurso mantido pelo servidor;
  - POST: permite que o cliente envie mensagens e conteúdo de formulários para servidores que irão manipular a informação de maneira adequada;
  - DELETE: permite que o cliente autorizado remova um recurso mantido pelo servidor.

# HTTP - GET





# HTTP - POST



# • Códigos de status das Respostas HTTP

- Os códigos de status das respostas HTTP indicam se uma requisição HTTP foi corretamente concluída.
- As respostas são agrupadas em cinco classes:
  - Respostas de informação (100-199),
  - Respostas de sucesso (200-299),
  - Redirecionamentos (300-399)
  - Erros do cliente (400-499)
  - Erros do servidor (500-599).

# • Códigos de status das Respostas HTTP

- Respostas informativas
- 100 Continue
  - Essa resposta provisória indica que tudo ocorreu bem até agora e que o cliente deve continuar com a requisição ou ignorar se já concluiu o que gostaria.
- 101 Switching Protocol
  - Esse código é enviado em resposta a um cabeçalho de solicitação Upgrade (en-US) pelo cliente, e indica o protocolo a que o servidor está alternando.
- 102 Processing (WebDAV (en-US))
  - Este código indica que o servidor recebeu e está processando a requisição, mas nenhuma resposta está disponível ainda.
- 103 Early Hints
  - Este código tem principalmente o objetivo de ser utilizado com o cabeçalho Link, indicando que o agente deve iniciar a pré-carregar (en-US) recursos enquanto o servidor prepara uma resposta.



# • Códigos de status das Respostas HTTP

- Respostas de sucesso

- GET: O recurso foi buscado e transmitido no corpo da mensagem.
- HEAD: Os cabeçalhos da entidade estão no corpo da mensagem.
- PUT ou POST: O recurso descrevendo o resultado da ação é transmitido no corpo da mensagem.
- TRACE: O corpo da mensagem contém a mensagem de requisição recebida pelo servidor.

# • Códigos de status das Respostas HTTP

- Respostas de sucesso
- 200 OK
  - Esta requisição foi bem sucedida. O significado do sucesso varia de acordo com o método HTTP:
- 201 Created
  - A requisição foi bem sucedida e um novo recurso foi criado como resultado. Esta é uma típica resposta enviada após uma requisição POST.
- 202 Accepted
  - A requisição foi recebida mas nenhuma ação foi tomada sobre ela. Isto é uma requisição não-comprometedora, o que significa que não há nenhuma maneira no HTTP para enviar uma resposta assíncrona indicando o resultado do processamento da solicitação. Isto é indicado para casos onde outro processo ou servidor lida com a requisição, ou para processamento em lote.
- 203 Non-Authoritative Information
  - Esse código de resposta significa que o conjunto de meta-informações retornadas não é o conjunto exato disponível no servidor de origem, mas coletado de uma cópia local ou de terceiros. Exceto essa condição, a resposta de 200 OK deve ser preferida em vez dessa resposta.
- 204 No Content
  - Não há conteúdo para enviar para esta solicitação, mas os cabeçalhos podem ser úteis. O user-agent pode atualizar seus cabeçalhos em cache para este recurso com os novos.

# • Códigos de status das Respostas HTTP

- Mensagens de redirecionamento
- 300 Multiple Choice
  - A requisição tem mais de uma resposta possível. User-agent ou o user deve escolher uma delas. Não há maneira padrão para escolher uma das respostas.
- 301 Moved Permanently
  - Esse código de resposta significa que a URI do recurso requerido mudou. Provavelmente, a nova URI será especificada na resposta.
- 302 Found
  - Esse código de resposta significa que a URI do recurso requerido foi mudada temporariamente. Novas mudanças na URI poderão ser feitas no futuro. Portanto, a mesma URI deve ser usada pelo cliente em requisições futuras.
- 303 See Other
  - O servidor manda essa resposta para instruir ao cliente buscar o recurso requisitado em outra URI com uma requisição GET.
- 304 Not Modified
  - Essa resposta é usada para questões de cache. Diz ao cliente que a resposta não foi modificada. Portanto, o cliente pode usar a mesma versão em cache da resposta.



# • Códigos de status das Respostas HTTP

- Respostas de erro do Cliente

- 400 Bad Request

- Essa resposta significa que o servidor não entendeu a requisição pois está com uma sintaxe inválida.

- 401 Unauthorized

- Embora o padrão HTTP especifique "unauthorized", semanticamente, essa resposta significa "unauthenticated". Ou seja, o cliente deve se autenticar para obter a resposta solicitada.

- 402 Payment Required

- Este código de resposta está reservado para uso futuro. O objetivo inicial da criação deste código era usá-lo para sistemas digitais de pagamento porém ele não está sendo usado atualmente.

- 403 Forbidden

- O cliente não tem direitos de acesso ao conteúdo portanto o servidor está rejeitando dar a resposta. Diferente do código 401, aqui a identidade do cliente é conhecida.

- 404 Not Found

- O servidor não pode encontrar o recurso solicitado. Este código de resposta talvez seja o mais famoso devido à frequência com que acontece na web.

- 405 Method Not Allowed

- O método de solicitação é conhecido pelo servidor, mas foi desativado e não pode ser usado. Os dois métodos obrigatórios, GET e HEAD, nunca devem ser desabilitados e não devem retornar este código de erro.

# • Códigos de status das Respostas HTTP

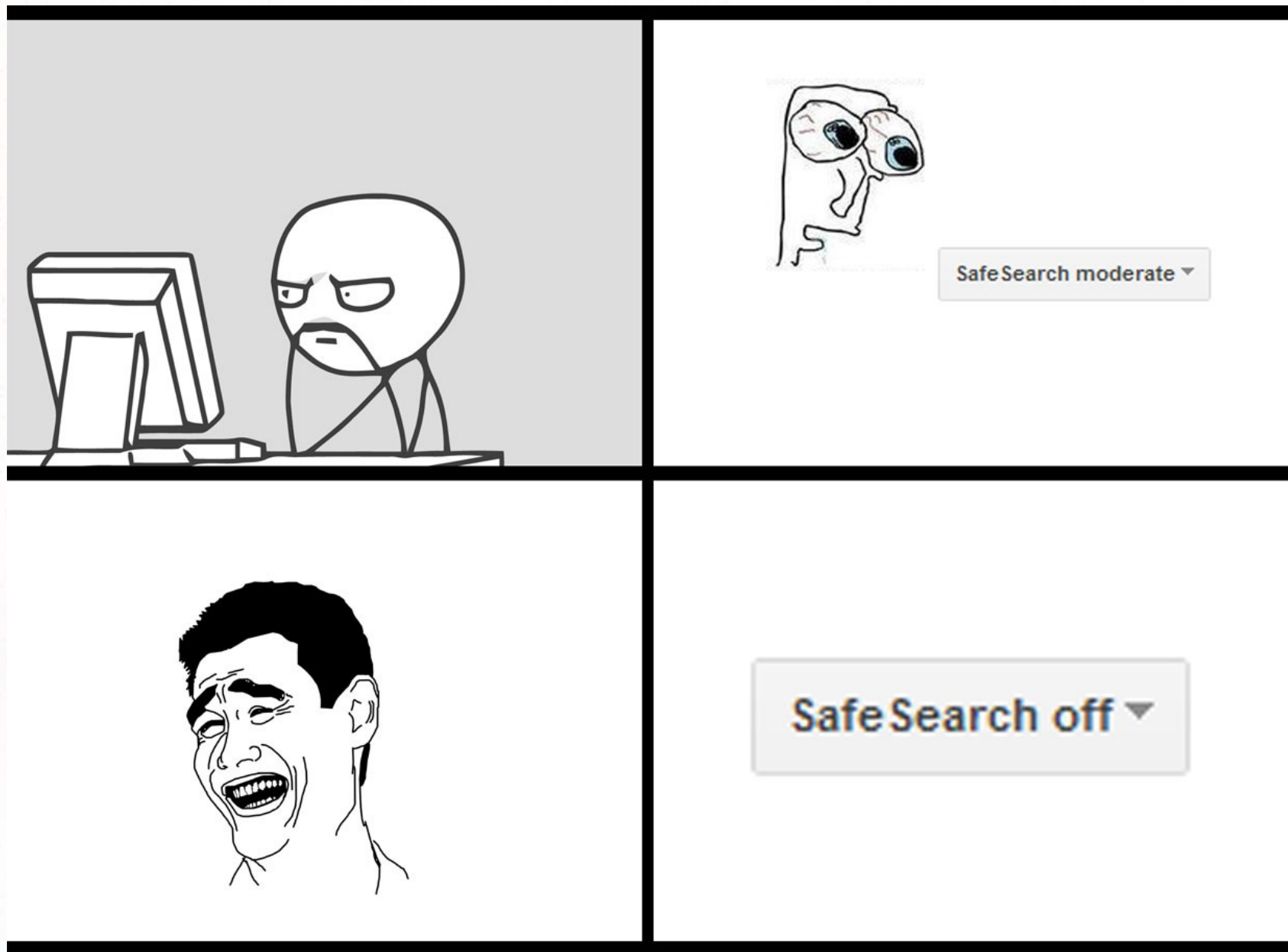
- Respostas de erro do Cliente
- 406 Not Acceptable
  - Essa resposta é enviada quando o servidor da Web após realizar a negociação de conteúdo orientada pelo servidor, não encontra nenhum conteúdo seguindo os critérios fornecidos pelo agente do usuário.
- 407 Proxy Authentication Required
  - Semelhante ao 401 porem é necessário que a autenticação seja feita por um proxy.
- 408 Request Timeout
  - Esta resposta é enviada por alguns servidores em uma conexão ociosa, mesmo sem qualquer requisição prévia pelo cliente. Ela significa que o servidor gostaria de derrubar esta conexão em desuso. Esta resposta é muito usada já que alguns navegadores, como Chrome, Firefox 27+, ou IE9, usam mecanismos HTTP de pré-conexão para acelerar a navegação. Note também que alguns servidores meramente derrubam a conexão sem enviar esta mensagem.
- 409 Conflict
  - Esta resposta será enviada quando uma requisição conflitar com o estado atual do servidor.
- 410 Gone
  - Esta resposta será enviada quando o conteúdo requisitado foi permanentemente deletado do servidor, sem nenhum endereço de redirecionamento. É esperado que clientes removam seus caches e links para o recurso. A especificação HTTP espera que este código de status seja usado para "serviços promocionais de tempo limitado". APIs não devem se sentir obrigadas a indicar que recursos foram removidos com este código de status.

# • Códigos de status das Respostas HTTP

- Respostas de erro do Servidor
- 500 Internal Server Error
  - O servidor encontrou uma situação com a qual não sabe lidar.
- 501 Not Implemented
  - O método da requisição não é suportado pelo servidor e não pode ser manipulado. Os únicos métodos exigidos que servidores suportem (e portanto não devem retornar este código) são GET e HEAD.
- 502 Bad Gateway
  - Esta resposta de erro significa que o servidor, ao trabalhar como um gateway a fim de obter uma resposta necessária para manipular a requisição, obteve uma resposta inválida.
- 503 Service Unavailable
  - O servidor não está pronto para manipular a requisição. Causas comuns são um servidor em manutenção ou sobrecarregado. Note que junto a esta resposta, uma página amigável explicando o problema deveria ser enviada. Estas respostas devem ser usadas para condições temporárias e o cabeçalho HTTP Retry-After: deverá, se possível, conter o tempo estimado para recuperação do serviço. O webmaster deve também tomar cuidado com os cabeçalhos relacionados com o cache que são enviados com esta resposta, já que estas respostas de condições temporárias normalmente não deveriam ser postas em cache.



# BUSCA ATIVA



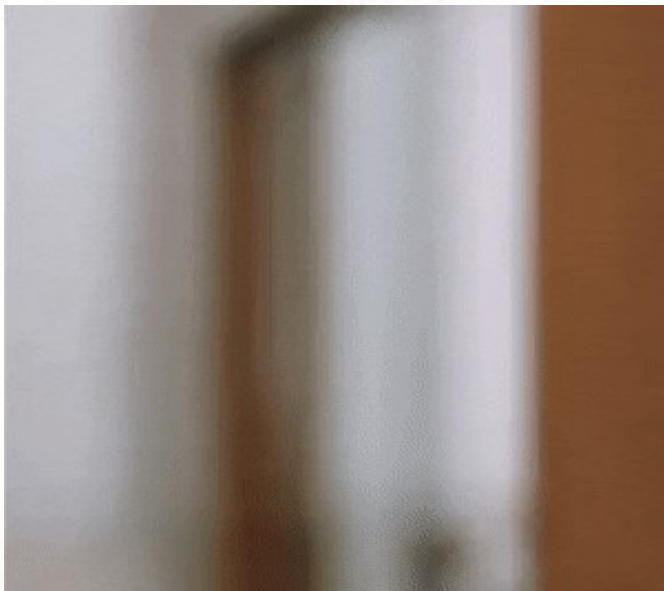
# Busca Ativa

- Pesquisar e entender como funciona...
  - FTP
  - SMTP
  - POP3
  - IMAP
  - SSH/TELNET

Dúvidas??







**Obrigado!**  
**Bom Dia!**  
**Boa Tarde!**  
**Boa Noite!**

ã

[gustavo.custodio@anhembib.com.br](mailto:gustavo.custodio@anhembib.com.br)



ecosistema  
ănimă