



Laboratório de Pesquisa em Redes e Multimídia

Redes de Computadores

Prof. José Gonçalves Pereira Filho
Departamento de Informática
zegonc@inf.ufes.br



Universidade Federal do Espírito Santo
Departamento de Informática

A faint, grayscale image of a computer keyboard is visible in the background, serving as a decorative element for the slide.

O Modelo OSI

(Aula 2)

O Modelo OSI

- *Framework* concebido pela ISO para a definição de padrões, visando a interconexão de sistemas heterogêneos (independência de fabricante, sistema operacional e plataforma de *hardware*).
- Descrito no documento ISO 7498, provê uma base conceitual para a interconexão de sistemas abertos e para o desenvolvimento de aplicações distribuídas.
- O termo “aberto” denota a habilidade do sistema em possuir conformidade com o modelo de referência e os padrões associados para a conexão.

As Camadas do Modelo

- O modelo separa as funcionalidades e as capacidades de arquitetura de rede em *camadas*. Define também os termos e objetos que são palavras reservadas no mundo das redes.
- Camadas foram escolhidas para quebrar o problema em problemas menores, de tamanho razoável, relativamente independentes.
- Cada camada (N) usa os serviços da camada inferior (N-1) e adiciona funcionalidades particulares para prover serviço à camada superior (N+1).

As Camadas do Modelo (cont.)

- As camadas definem desde aspectos físicos até aspectos abstratos da aplicação.
- O modelo OSI é constituído de sete camadas: *Aplicação, Apresentação, Sessão, Transporte, Rede, Enlace e Física.*

As Camadas do Modelo (cont.)

Aplicação

Apresentação

Sessão

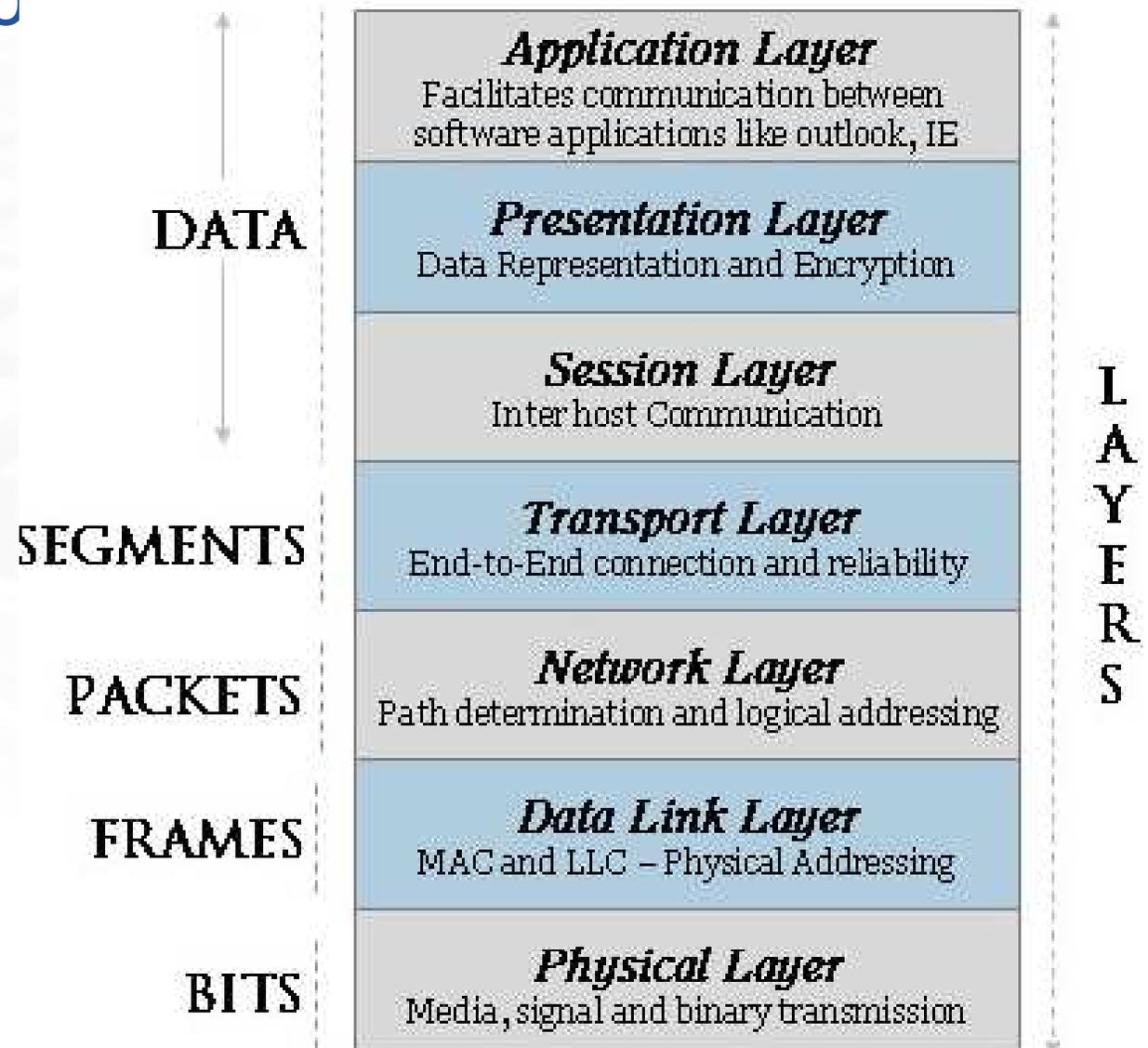
Transporte

Rede

Enlace

Física

As Camadas do Modelo (cont.)



As Camadas do Modelo (cont.)

- Camadas superiores
 - Prestam serviços relacionados com a natureza da aplicação. Tratam de aspectos de *interoperação de aplicações*.
- Camadas inferiores
 - Possibilitam a interconexão de sistemas ou equipamentos individuais. Estão relacionadas a aspectos de *transmissão e interconexão*.
- Camada de transporte
 - Provê *comunicação fim-a-fim* entre aplicações.

Princípios do Modelo OSI

- Agrupar funções similares em uma mesma camada, isto é, cada camada deve desempenhar uma função bem definida.
- Criar camadas separadas para manipular funções que são manifestamente diferentes no processo ou na tecnologia envolvida.
- Permitir alterações de funções ou protocolos dentro de uma camada sem afetar as outras.
- Uma camada deve ser criada onde houver necessidade de um outro grau de abstração.
- Não criar um número grande de camadas para que a tarefa de descrever e integrar as camadas não fique mais complexa do que o necessário.

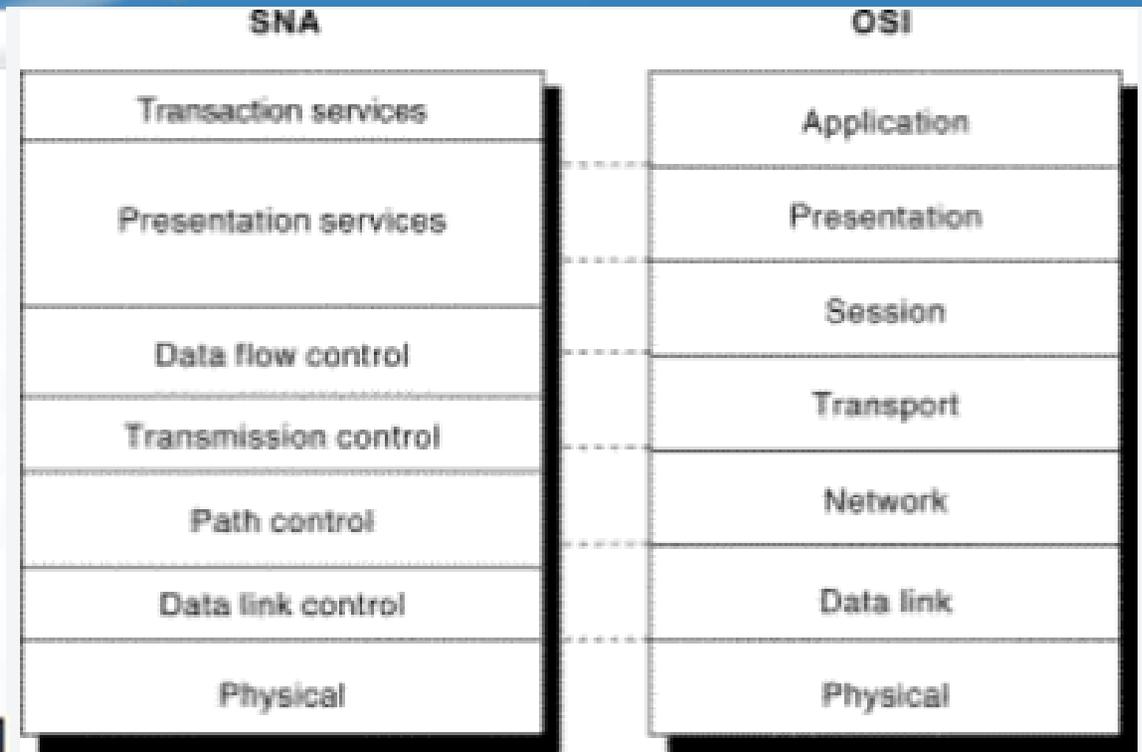
Princípios do Modelo OSI (cont.)

- Criar, para cada camada, fronteiras somente com a sua camada superior e inferior.
- As fronteiras entre camadas devem ser escolhidas de forma a minimizar o fluxo de informações através das interfaces, ou seja, criar fronteiras em pontos onde a descrição dos serviços possa ser pequena e o número de interações através da fronteira seja minimizado.
- Criar uma fronteira onde a experiência do passado tenha demonstrado ser necessária essa separação.

Princípios do Modelo OSI (cont.)

- Em suma, o número de camadas deve ser grande o bastante para que funções distintas não precisem ser desnecessariamente colocadas na mesma camada e pequeno o suficiente para que a arquitetura não se torne difícil de controlar.

Outros Exemplos de Arquiteturas de Redes em Camadas

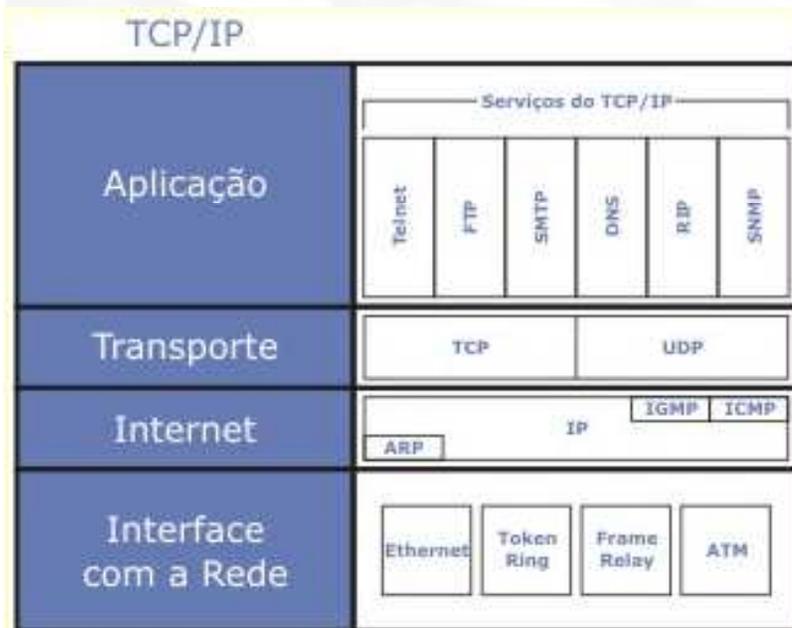


7	Aplicação		Aplicação
6	Apresentação		
5	Sessão		
4	Transporte		Transporte
3	Rede		Internet
2	Enlace de Dados		Interface com a Rede
1	Conexão Física		

Modelo OSI

TCP/IP

Outros Exemplos de Arquiteturas de Redes em Camadas (cont.)



OSI Reference Model

AppleTalk Protocols

7 Application

6 Presentation

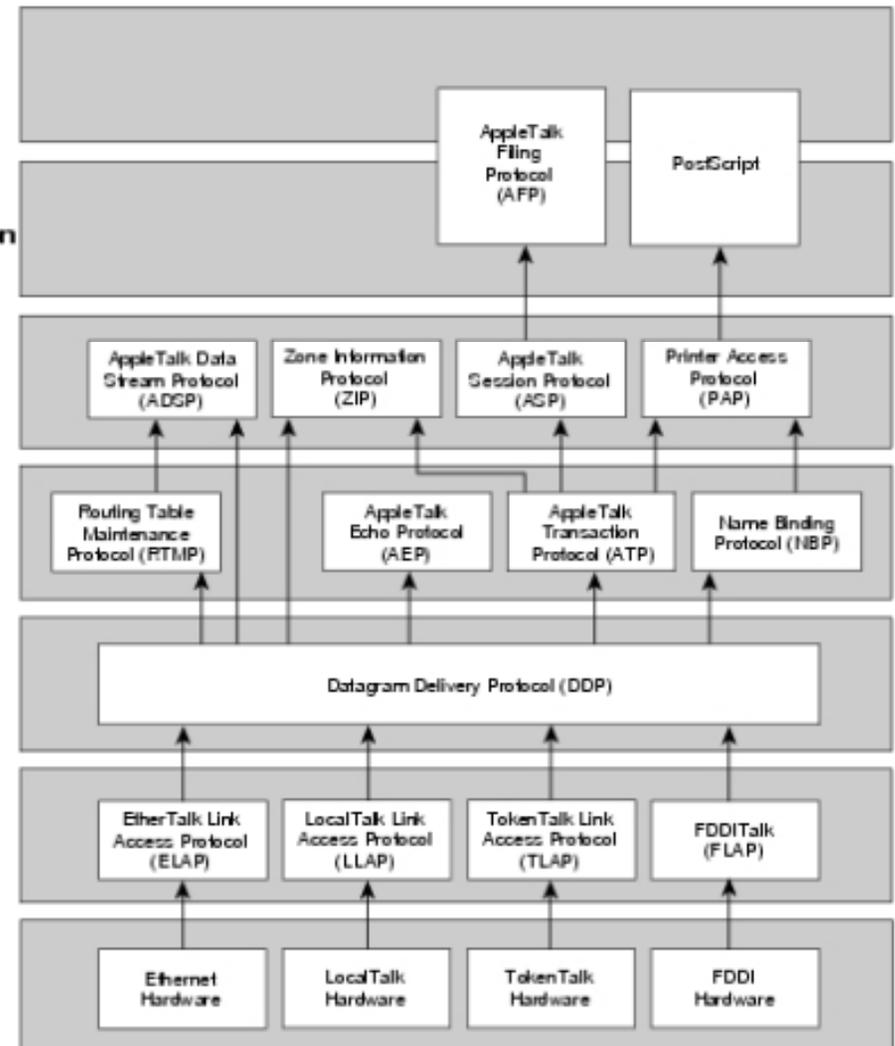
5 Session

4 Transport

3 Network

2 Data Link

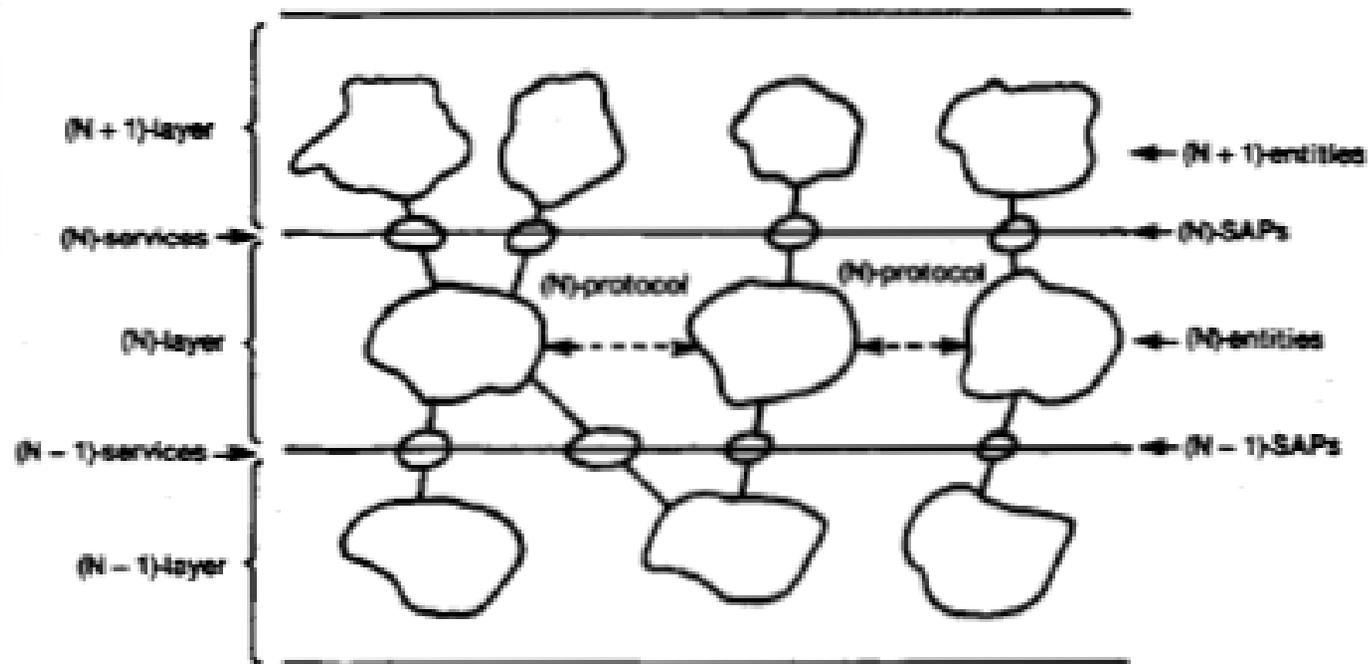
1 Physical



Entidades da Camada

- No MR OSI as camadas parceiras se comunicam através de um objeto chamado *entidade da camada*.
- Entidade é um termo que significa uma capacidade de comunicação (por exemplo, protocolo IP, protocolo TCP, um elemento roteador, etc).
- As entidade podem ser elementos de *software* ou de *hardware*.

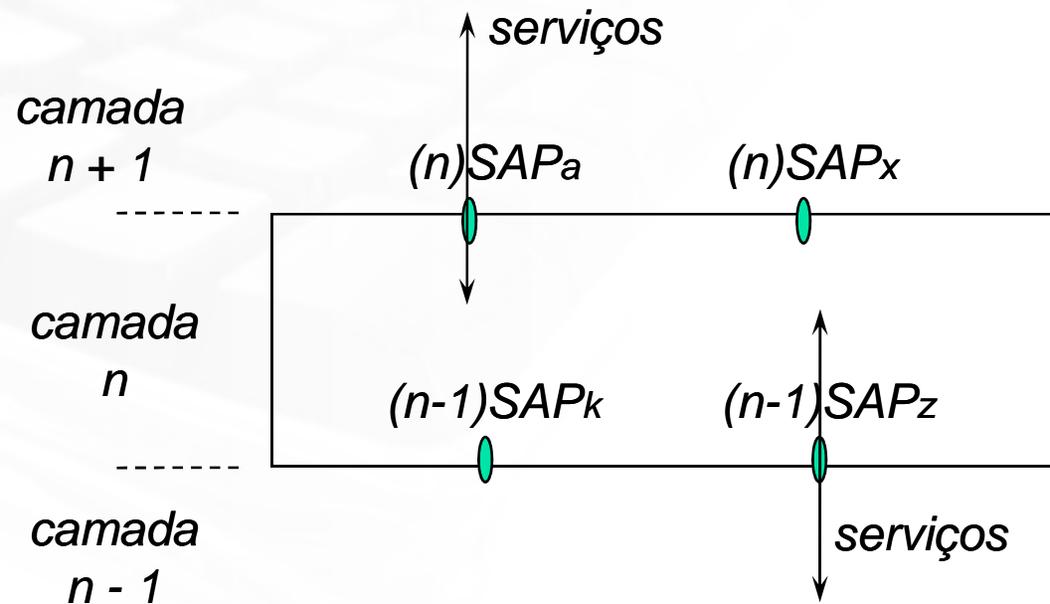
Entidades da Camada (cont.)



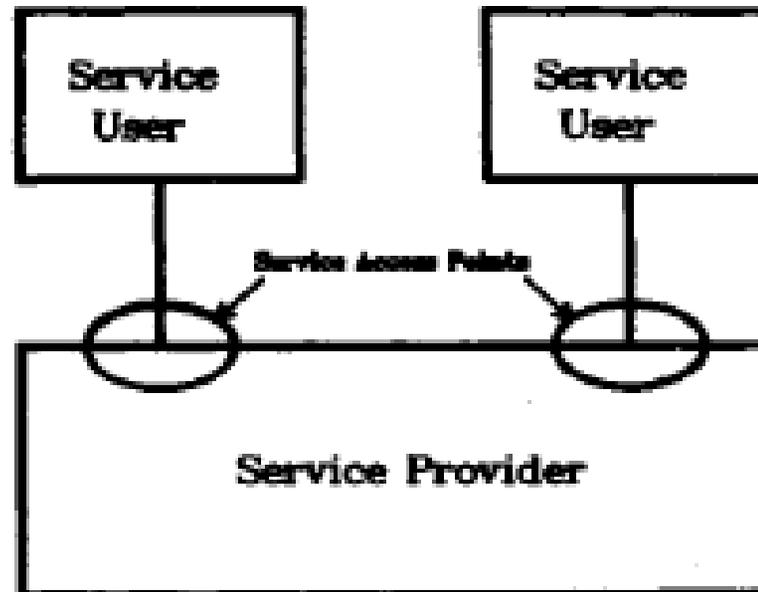
Serviços

- A comunicação entre camadas é feita através da requisição de (e da resposta a) *serviços*. Cada camada é responsável por um conjunto de serviços
 - Serviço = *o que*
- Serviços são solicitados (respondidos) através de pontos específicos localizados nas interfaces entre as camadas, denominados de *Pontos de Acesso a Serviços (SAP's - Service Access Points)*.
- **A prestação de serviços é o que justifica a existência de uma camada.**
- Uma camada (N+1) requisita serviços a uma camada (N) através da invocação de *primitivas de serviço* (ex: *connect, abort, data*).

Serviços (cont.)



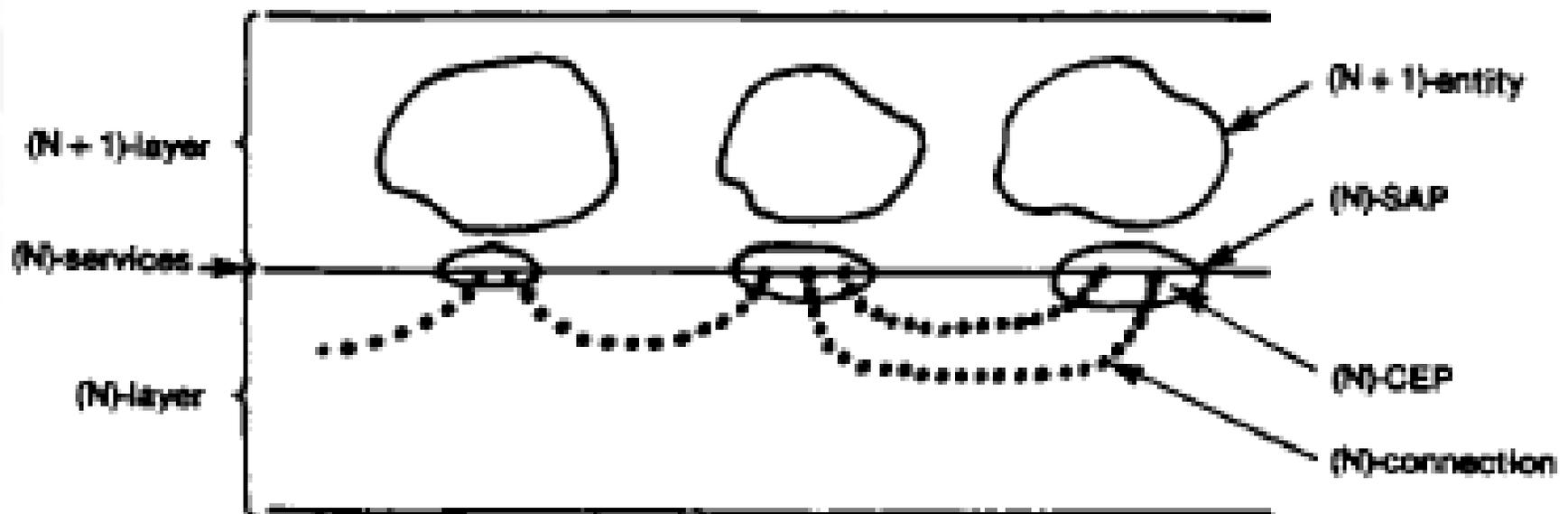
Serviço (cont.)



Conexão

- Um serviço comum oferecido por todas as camadas consiste em prover associações entre SAP's, que podem ser usados em, particular, para transferir dados.
- Mais precisamente, uma *camada-(N)* oferece *conexões-(N)* entre *SAP's-(N)* como parte de *serviços-(N)*. Essas conexões podem ser *unicast*, *multicast* ou *broadcast*.
- O final de uma conexão-(N) em um SAP-(N) é chamado de CEP-(N) ("*connection-end-point*").

Conexão (cont.)

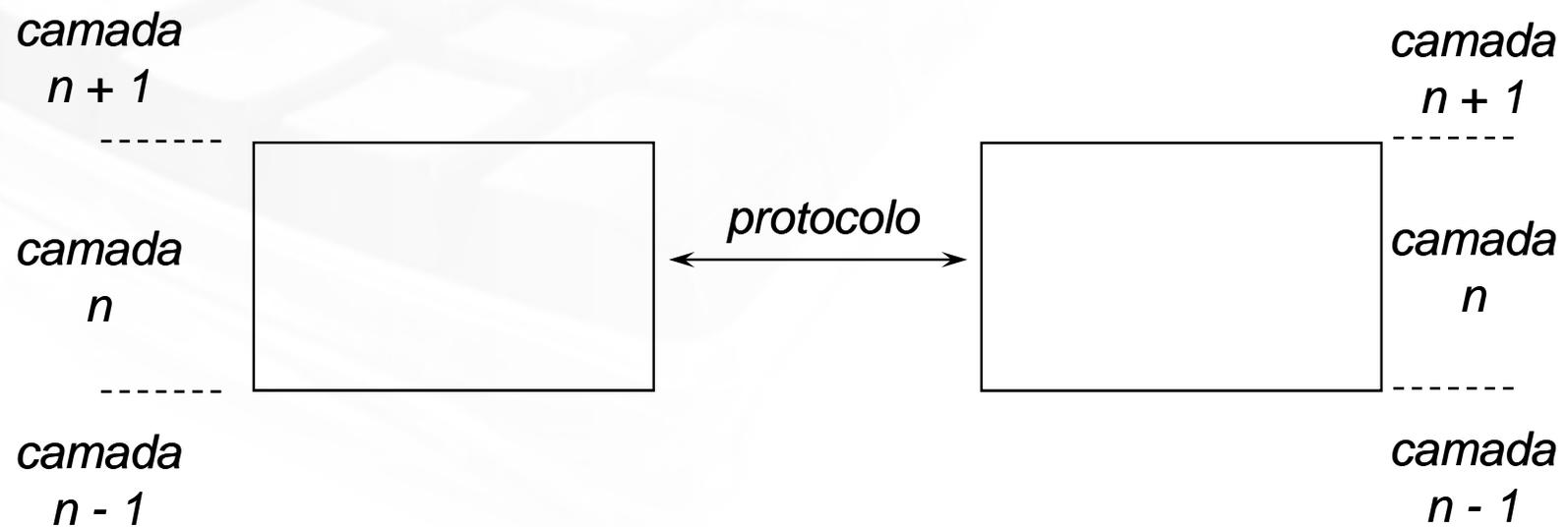


Protocolos

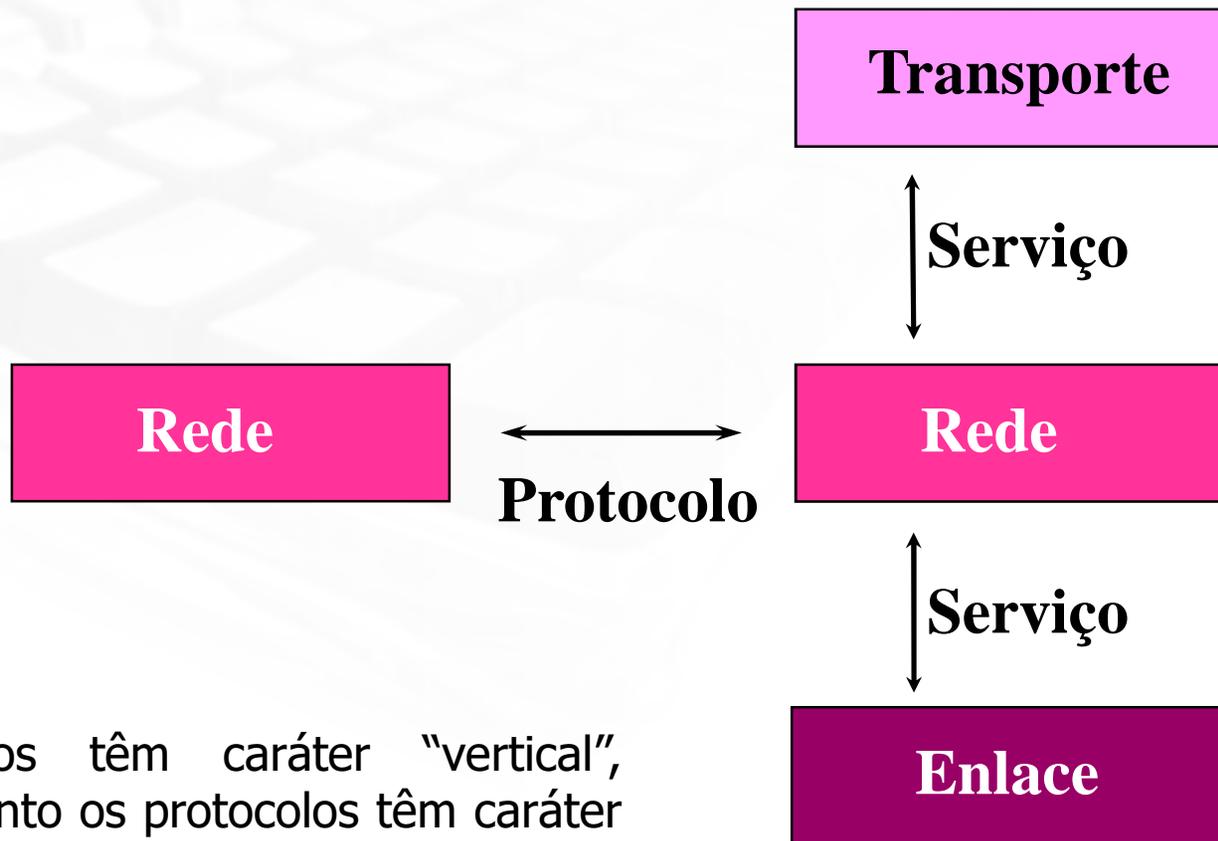
- Protocolos são um conjunto de regras que governa a interação em sistemas distribuídos.
- A comunicação entre camadas de mesmo número em nós distintos é feita através de *protocolos*.
 - Para que dois parceiros se comuniquem eles devem especificar o mesmo protocolo.
- Os protocolos existem como forma de viabilizar a prestação de serviços pelas camadas (protocolo = *como*).

Protocolos (cont.)

- Comunicação entre camadas parceiras.



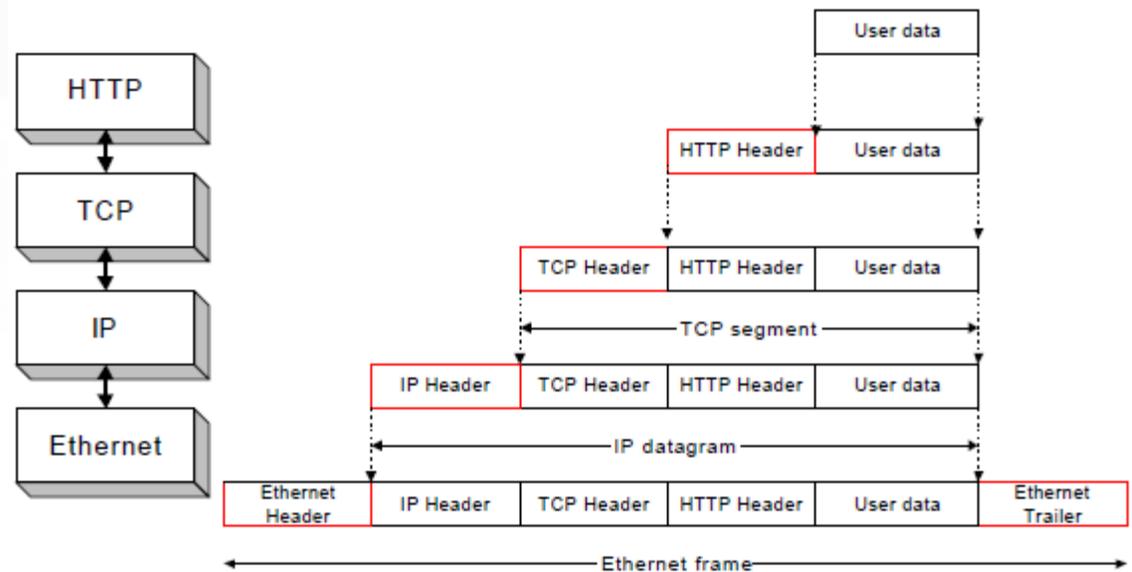
Serviço vs. Protocolo



Serviços têm caráter "vertical", enquanto os protocolos têm caráter "horizontal".

Encapsulamento de Dados

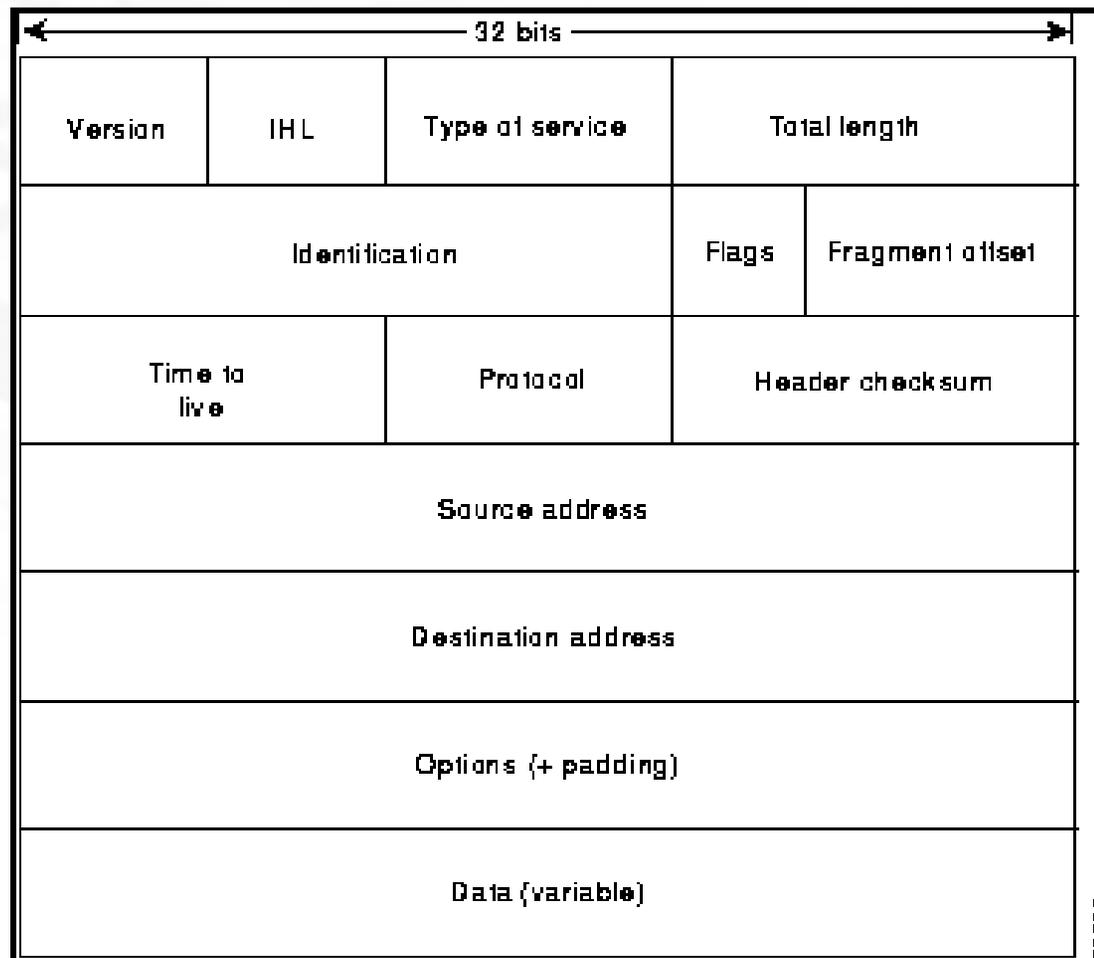
- Protocolos da camada N adicionam informações às unidades de dados recebidas da camada (N+1), gerando dados de maior tamanho a cada camada do modelo.
- Os serviços de uma camada recebem o respectivo protocolo e são passados, através do SAP, à camada inferior.



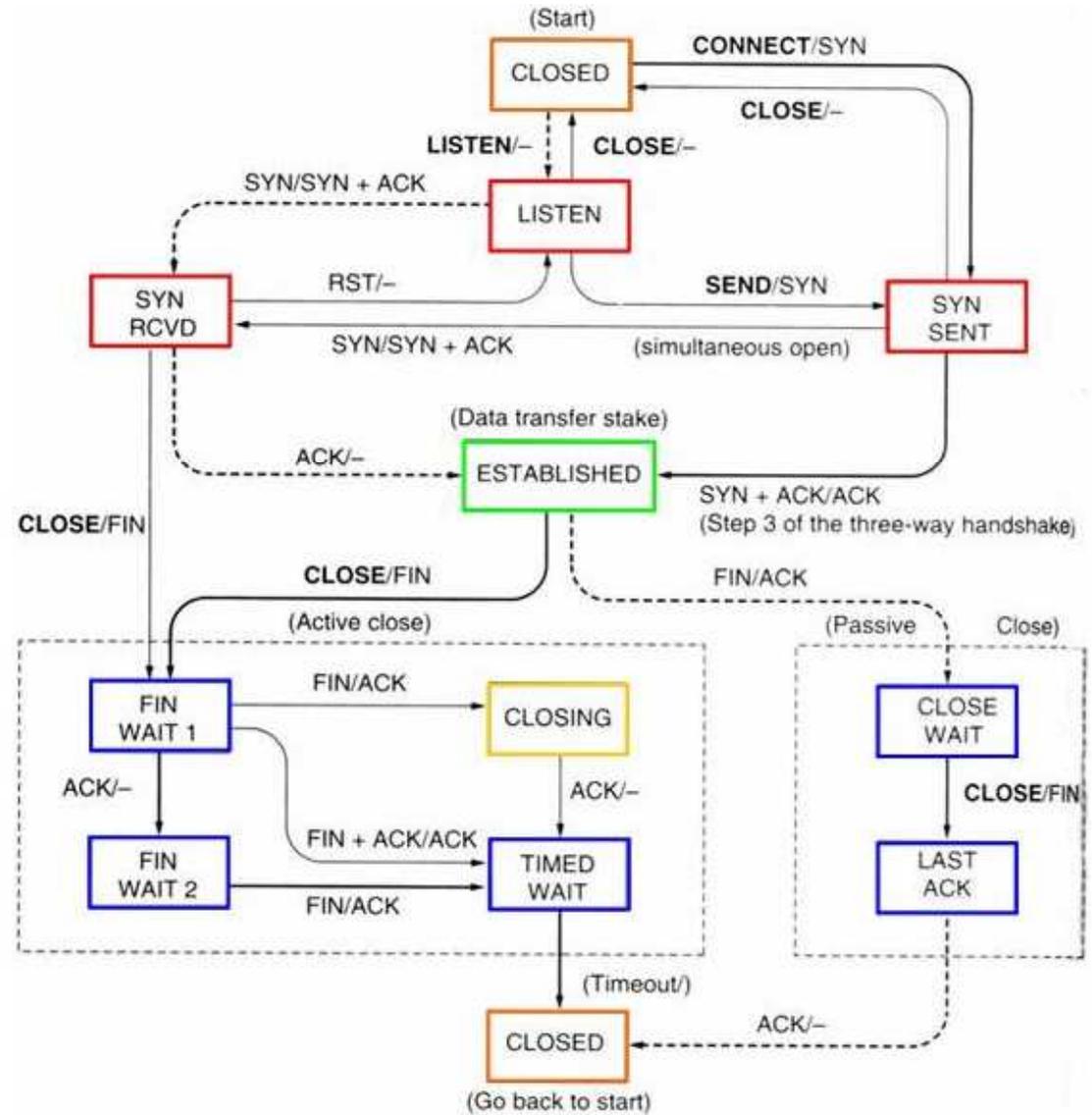
Elementos de um Protocolo

- **Sintaxe:**
 - Inclui aspectos como formato dos dados e níveis de sinal.
- **Semântica:**
 - Inclui informação de controle para coordenação e manipulação de erros.
- **Temporização:**
 - Inclui aspectos temporais envolvidos na troca de dados entre transmissor e receptor.

Sintaxe: Cabeçalho do Protocolo IP



Semântica: Máquina de Estados do Protocolo TCP



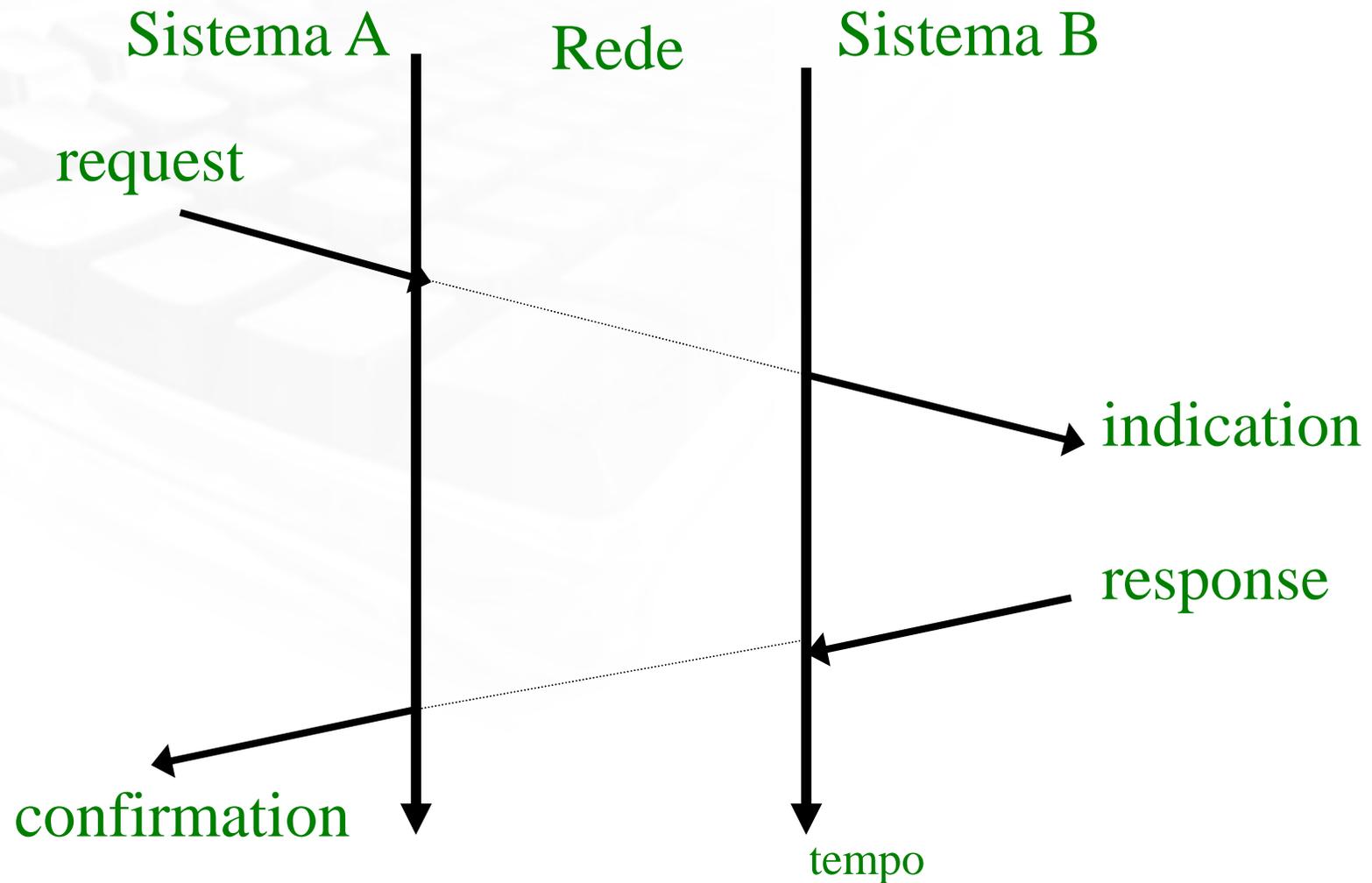
Aspectos de Temporização: Estabelecimento de Conexão

Tempo	Evento	Diagrama
	O computador A envia um pacote de sincronismo (SYN) para o computador B	<pre>sequenceDiagram participant A as HOST A participant B as HOST B Note over A: t A->>B: syn Note over B: t+1 B->>A: ack syn Note over A: t+2 A->>B: ack Note over B: t+3 B->>A: Note over A: t+4 A->>B: Note over B: t+5</pre>
+1	O computador B recebe o pacote (SYN) do computador A	
+2	O computador B envia seu próprio pacote de sincronismo (SYN) e o reconhecimento (ACK)	
+3	O computador A recebe o pacote SYN de B	
+4	O computador A envia o seu pacote de reconhecimento positivo (ACK)	
+5	O computador B recebe o ACK, e finalmente a conexão TCP é estabelecida e a transmissão dos pacotes de dados é iniciada até finalizar a sessão TCP.	

Primitivas de Serviço

- Do ponto de vista abstrato, as primitivas de serviço podem ser:
 - *Requisição*: quando um serviço é requisitado para ser desempenhado no parceiro.
 - *Resposta*: quando, uma vez desempenhado pelo parceiro, é gerada uma resposta ao serviço requisitado.
- As primitivas podem conter duas situações possíveis:
 - Primitivas de Requisição:
 - No instante em que é enviada para a rede: "*request*".
 - No instante que a requisição chega no parceiro: "*indication*".
 - Primitivas de Resposta:
 - No instante em que é enviada: "*response*".
 - No instante que chega no requisitante: "*confirmation*".

Representação das Primitivas



Grupos de Serviços

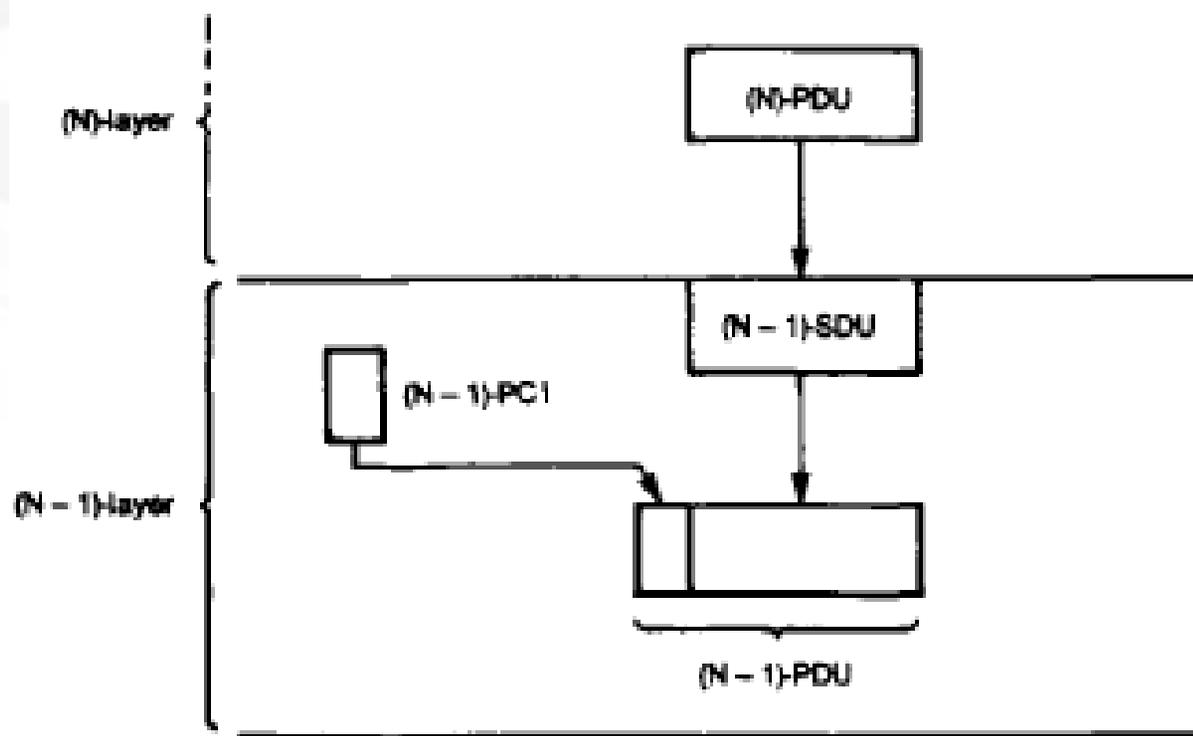
- Serviços confirmados:
 - São serviços que contém as quatro fases da primitiva (*request*, *indication*, *response*, *confirmation*).
- Serviços não-confirmados:
 - São serviços que especificam apenas as fases de *request* e *indication*.

PDU e SDU

- Quando uma camada (N+1) requisita um serviço à camada (N), neste instante ela está enviando um conjunto de bytes, denominado *Protocol Data Unit* (PDU).
- A PDU está assim dividida:
 - Cabeçalho: a parte do protocolo da camada (N+1);
 - Conteúdo: a parte de dados da camada (N+1).
- A PDU da camada (N+1) se encaixa na parte de dados da camada (N).
- Assim que a (N)-PDU ultrapassa a fronteira entre as camadas (N) e (N-1) ela recebe um novo nome na camada (N-1): SDU (*Service Data Unit*).

PDU e SDU (Cont.)

- $(n-1)\text{PDU} = (n-1)\text{SDU} + (n-1)\text{PCI}$



PCI = Protocol-control-information

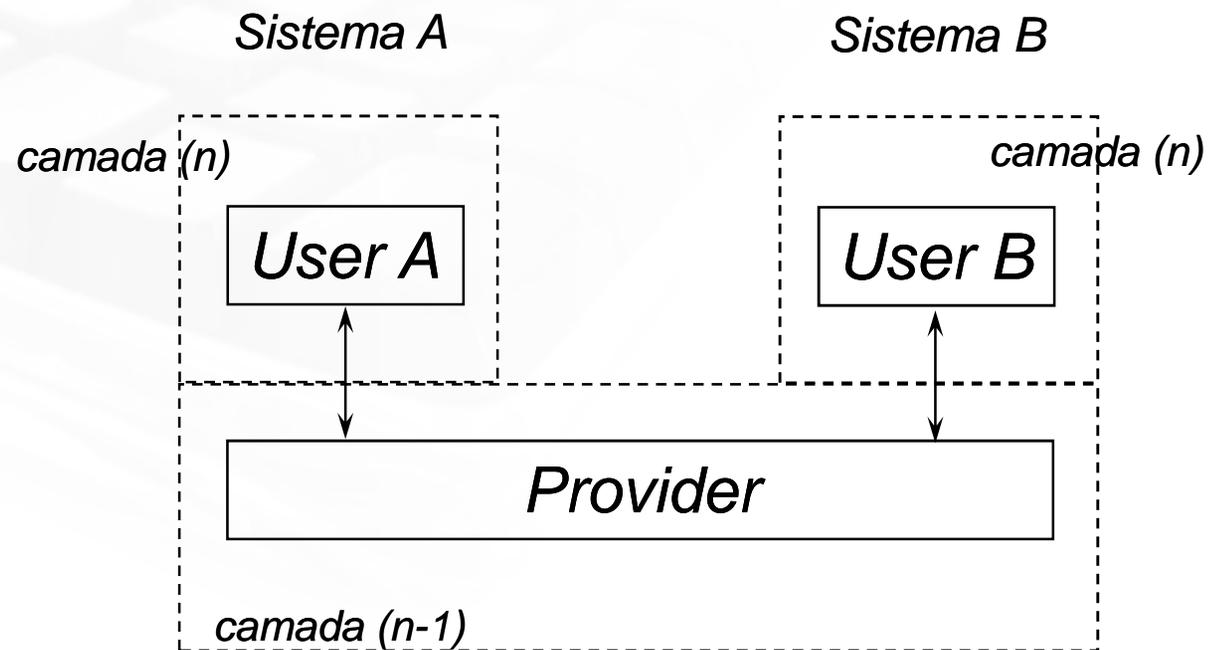
PDU = Protocol-data-unit

SDU = Service-data-unit

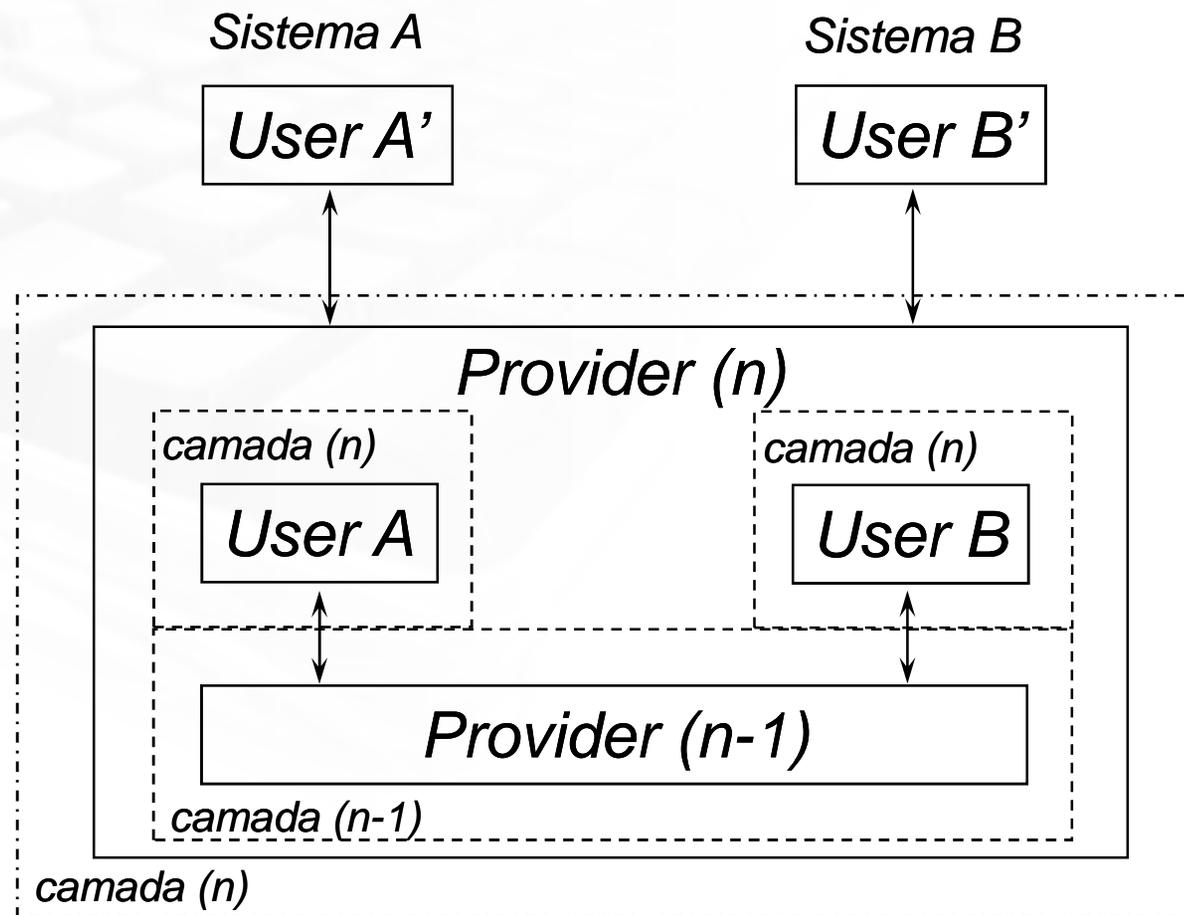
Camadas *User* e *Provider*

- A camada (N) requisita serviços somente da camada imediatamente inferior (N-1); por essa razão, ela é dita usuária (*user*) dessa camada.
- Uma camada abstrai a existência das camadas mais inferiores, oferecendo a somatória das funcionalidades de todas as camadas inferiores.
- Por essa abstração, a camada (N) é dita provedora (*provider*) de serviços para a camada superior (N+1).
- O provimento de serviços abstrai, inclusive, o aspecto da comunicação com a camada parceira.
- **Portanto, o *provider* oferece os *serviços* e a *conexão* da camada (N-1), a um usuário da camada (N).**

Camadas *User* e *Provider* (cont.)



Visão Geral



Por que o OSI não vingou?

- Segundo Tanenbaum, alguns dos motivos foram:
 - *Momento ruim*: Quanto o padrão OSI foi lançado, o TCP/IP já estava sendo amplamente utilizado nas universidades. Muitos fabricantes já ofereciam produtos TCP/IP. Desta forma, não havia interesse em gastar dinheiro em uma segunda pilha de protocolos.
 - *Tecnologia Ruim*: A divisão em 7 camadas foi mais política que técnica. Duas delas encontram-se praticamente vazias (sessão e apresentação) e outras duas sobrecarregadas (enlace e rede). Muitas das funções de uma camada, como o controle de erros, apareciam repetidas em outras. Se tudo isso não bastasse, a implementação era bastante complexa.
 - *Implementação Ruim*: Devido à tecnologia ruim, as implementações eram lentas.
 - *Política Ruim*: O OSI era visto como algo sendo empurrado por burocratas a pesquisadores. Por outro lado, o TCP/IP estava associado ao UNIX, que era adorado no ambiente acadêmico.