

Modelo de Referência OSI

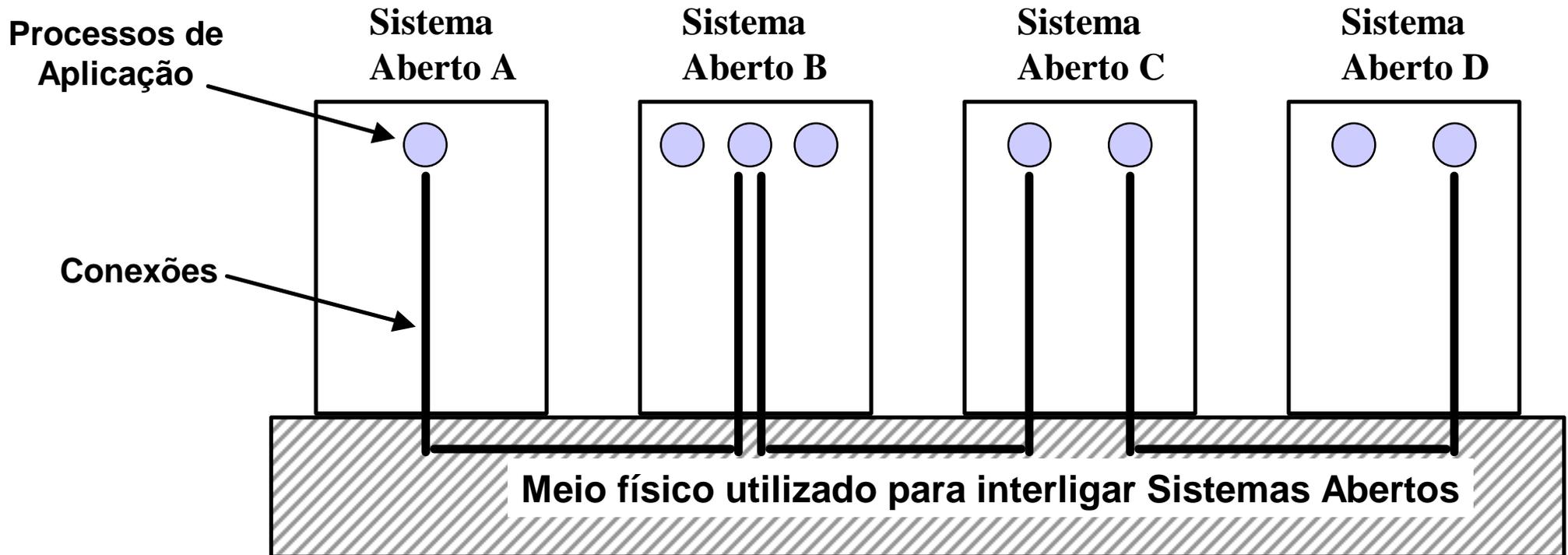
Hermes Senger

*Pós-Graduação “Lato Sensu” em Redes de
Computadores - DC - UFSCar*

A necessidade de padronização

- Década de 70 :
 - Sucesso das primeiras redes de dados (ARPANET, Cyclades, etc)
 - Com o processo de barateamento dos dispositivos de hardware, alguns fabricantes começaram a produzir equipamentos baseados em protocolos proprietários de comunicação de dados⇒ **A comunidade de usuários logo percebeu a necessidade de se ter padrões, pois a maioria das organizações possuía equipamentos de diferentes fabricantes.**
 - E para que nenhum dos fabricantes levassem vantagem, esses padrões deveriam ser **abertos**, ou seja, independentes de fabricante.
- Assim, em 1978 a ISO (“International Organization for Standardization ”) Comitê Técnico 97 (Processamento de Informação) decidiu criar Subcomitê 16 (SC16) para estudar padrões para sistemas abertos. A ISO é uma organização internacional, criada em 1946, para definir e divulgar padrões. A ISO é composta de diversas organizações nacionais de diversos países. A ABNT (Associação Brasileira de Normas Técnicas) é quem representa o Brasil junto à ISO.

Interconexão de sistemas abertos



- Este é o modelo criado para suportar a troca de informações entre **processos de aplicação** ou **entidades de aplicação**.

O modelo OSI

- Assim foi criado o padrão internacional 7498, denominado **Open Systems Interconnection**, que define um modelo de referência para a interconexão de sistemas abertos.
- Modelo OSI define **um conjunto de 7 camadas** e de **funções a serem executadas em cada camada**. Esse número (sete) não é propriamente um número mágico, mas é um compromisso entre **gerenciabilidade** e **desempenho** (poucas camadas significa acúmulo de funções na mesma camada e maior dificuldade de implementação, e muitas camadas pode implicar em dificuldade de gerenciar um número grande de camadas).
- **Objetivos:**
 - criar um modelo padrão, baseado em uma arquitetura de diversos níveis que possa orientar o projeto e implementação de protocolos.
 - O objetivo principal é a **convergência de sistemas**.
 - Deve tratar o problema em diversos níveis, especificando o que deve ser tratado por cada um dos níveis, bem como as **interfaces** entre cada nível. Assim, se forem feitas alterações no âmbito de um determinado nível, **isto não deve afetar a estrutura global**.

Uma arquitetura em camadas

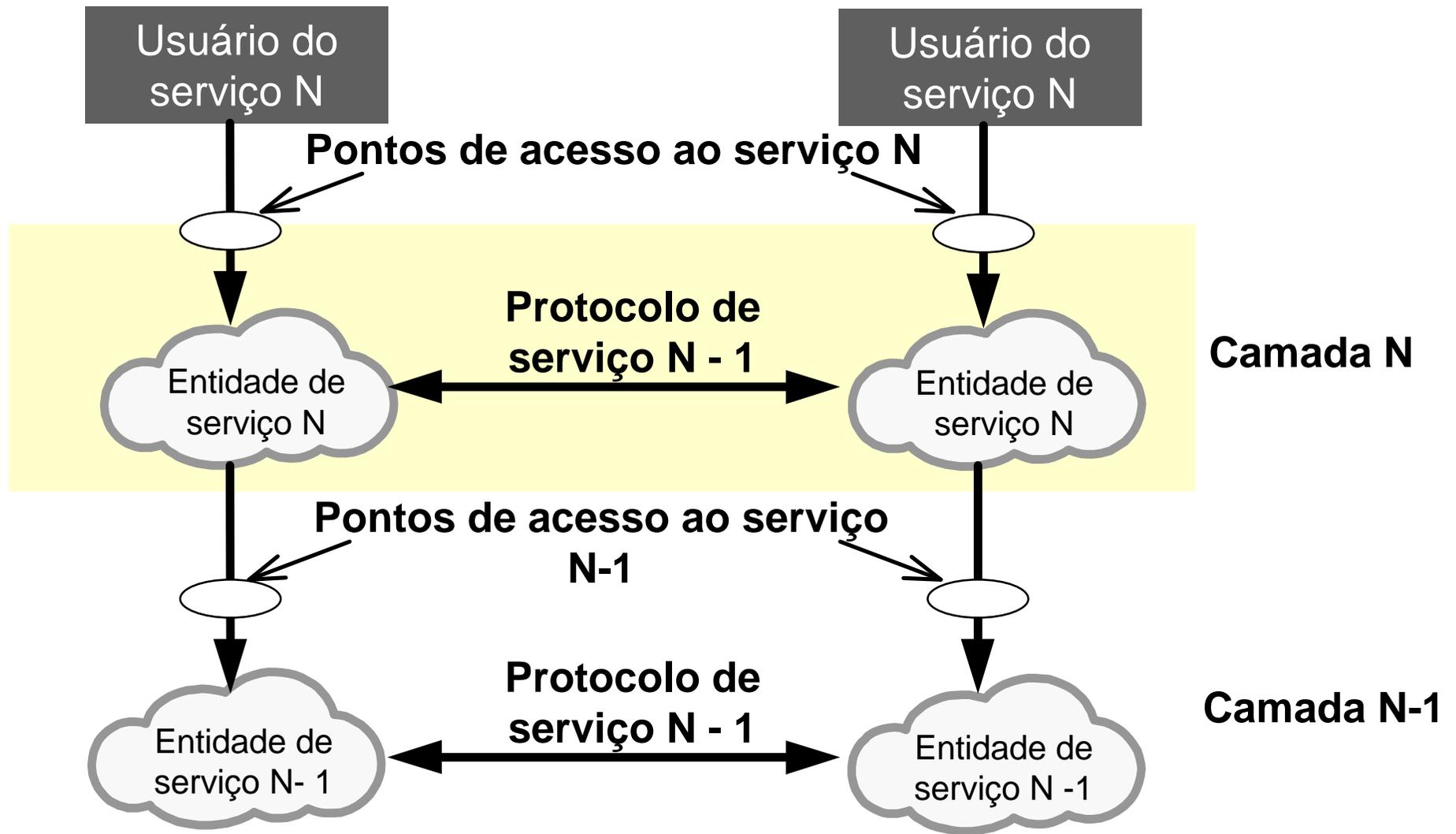


Meio físico utilizado na interconexão

Modelo OSI x Arquitetura

- Note que o **modelo** OSI não define propriamente uma **arquitetura** de rede, pois ele não especifica com exatidão os serviços e protocolos de cada camada [Soares, 95]. Ele apenas define funções, ou seja, “o que cada camada deve fazer”.
- Assim, o fato de que dois sistemas abertos utilizam o modelo OSI não garante que possam trocar dados entre si, pois poderiam estar utilizando diferentes serviços ou protocolos em cada uma das camadas.
- Para resolver isso, a ISO foi adiante e definiu o que chama de *perfis funcionais*, ou seja, grupos de opções de serviço/protocolos padronizados que, quando escolhidos igualmente por dois sistemas abertos, garantem sua interoperabilidade. Para ver mais detalhes sobre perfis funcionais veja [Soares 95].

Princípio de funcionamento do modelo OSI

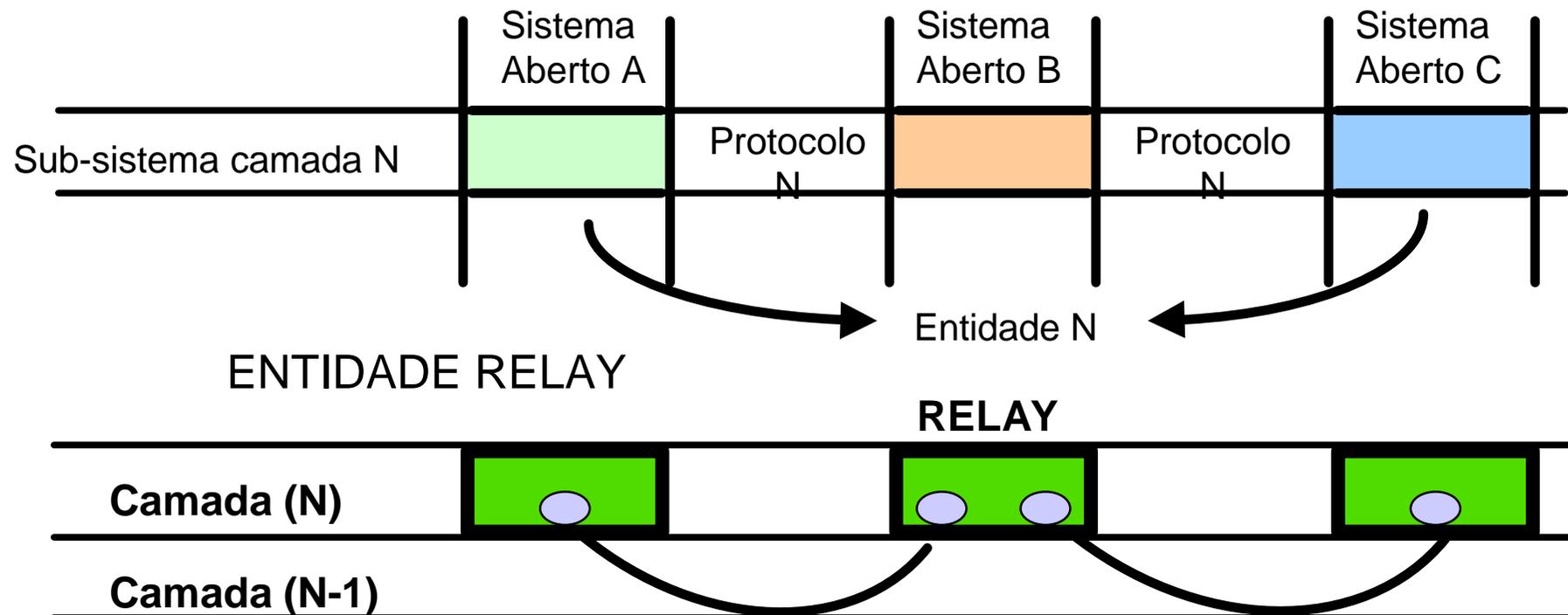


Terminologia adotada no modelo OSI

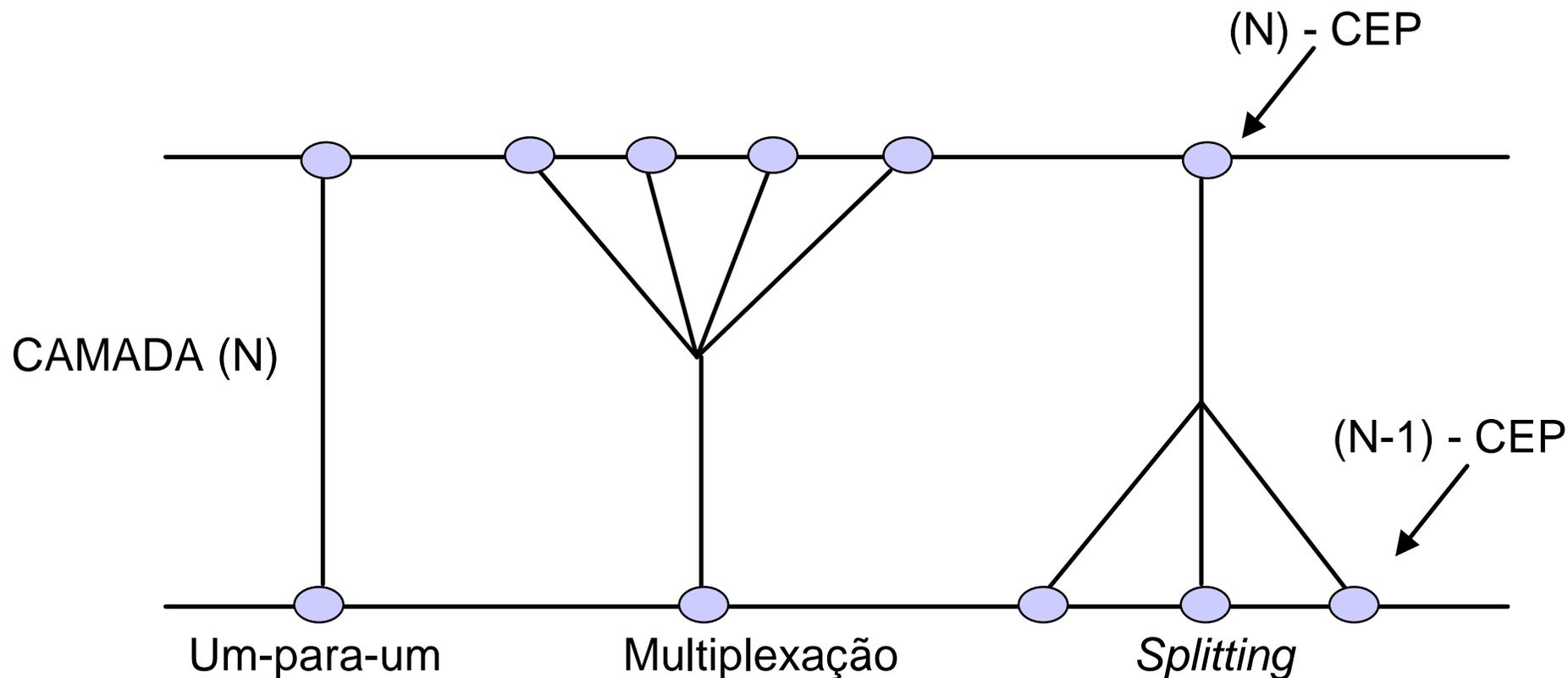
- Um **serviço** representa um conjunto de funcionalidades prestadas e oferecidas a um **usuário** por um **fornecedor**.
- Um **usuário** só pode acessar um **serviço** através de um ponto de **acesso ao serviço (SAP - service access point)**. Os **SAPs** da camada **N** são os locais onde os **usuários** da camada **N+1** podem acessar os **serviços** da camada **N**. Cada **SAP** tem um endereço único que o identifica.
- As **entidades** de uma camada são elementos de *hardware* (por exemplo, uma placa de rede) ou *software* (processos) que atuam naquela camada. **Entidades** de uma mesma camada, mas que residem em máquinas diferentes são chamadas de **entidades pares**.

COMUNICAÇÃO ENTRE ENTIDADES

- **Entidades pares** podem comunicar-se diretamente, ou através de intermediários, denominados **entidades relay**.



Relações entre conexões de camadas adjacentes



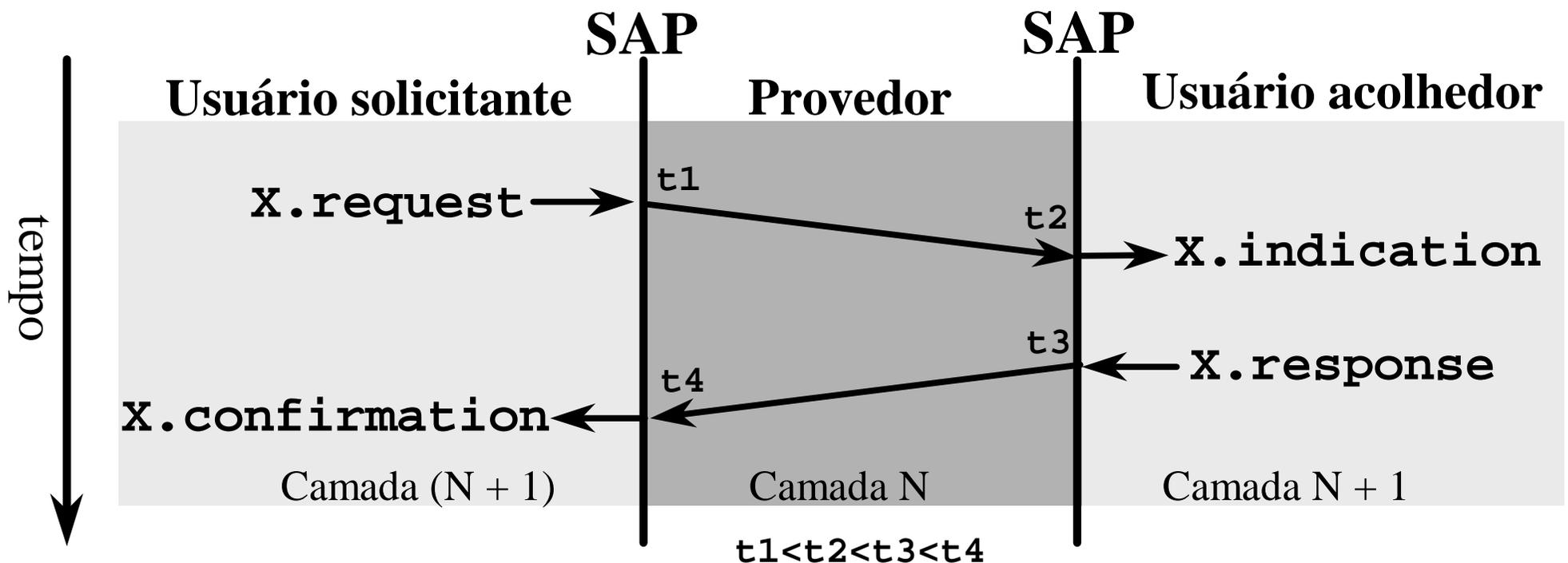
CEP - Connection End Point

Terminologia adotada no modelo OSI

- Um serviço é fornecido por uma (**N**) camada a outra camada (**N+1**) através das **primitivas de serviço** trocadas entre elas.
- Basicamente, existem 4 tipos de primitivas:
 - **x.REQUEST**
 - enviada pelo usuário solicitante
 - **x.INDICATION**
 - entregue pelo fornecedor do serviço ao usuário que aceita o serviço
 - **x.RESPONSE**
 - invocada pelo usuário que aceita o serviço
 - **x.CONFIRMATION**
 - entregue ao solicitante pelo fornecedor do serviço
- Obs. **x** deve indicar o tipo de serviço que está sendo solicitado.

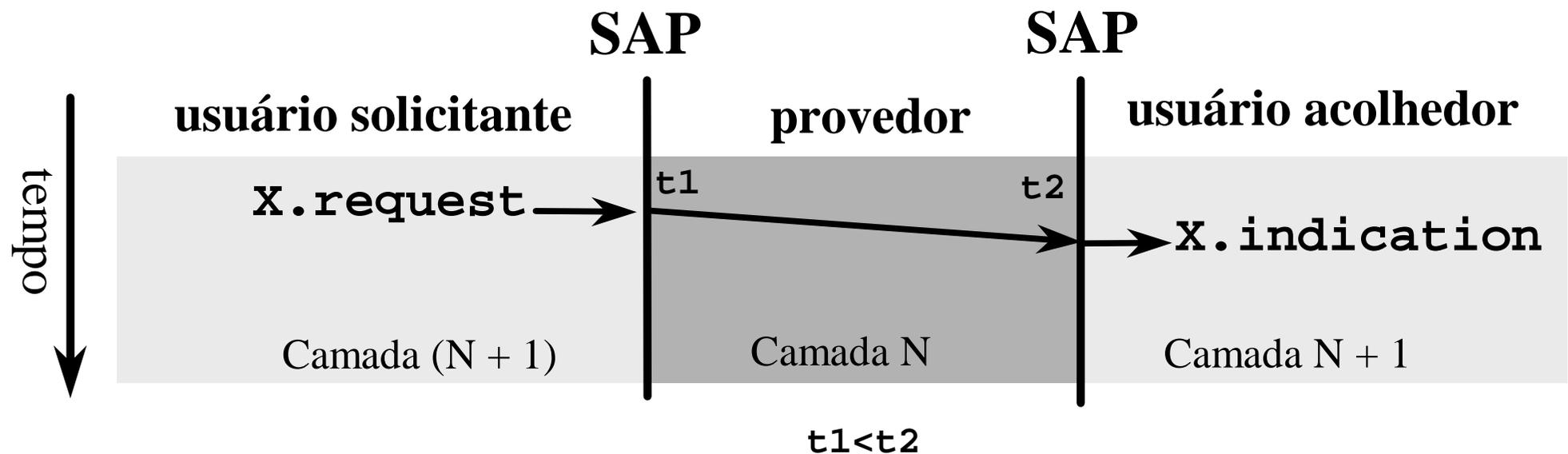
Serviços com confirmação

- Um serviço pode ser **confirmado** ou **não-confirmado**.
- Nos serviços **confirmados**, é preciso haver um acordo entre o usuário **solicitante** e o usuário que aceita o serviço (**acolhedor**). Nesse caso, o usuário **acolhedor** pode aceitar ou rejeitar o serviço (X).



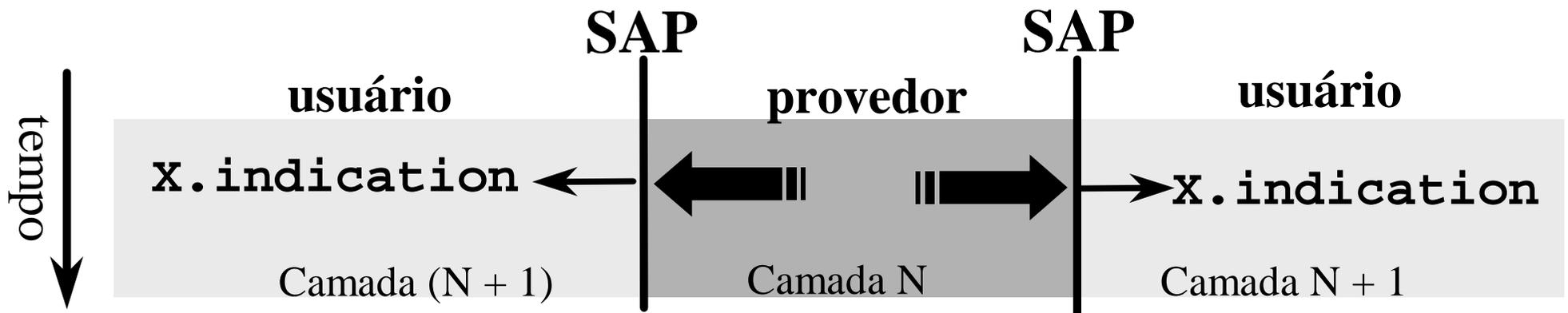
Serviços sem confirmação

- Nos serviços **não-confirmados** ou **sem confirmação**, não há necessidade de haver um acordo entre o usuário **solicitante** e o usuário **acolhedor**.



Serviços iniciado pelo fornecedor

- Existe ainda, um tipo de serviço **iniciado pelo fornecedor** (no exemplo, a camada N). Neste caso, o **fornecedor (ou provedor)** entrega a ambos usuários, a primitiva `x.indication` em resposta a algum evento interno

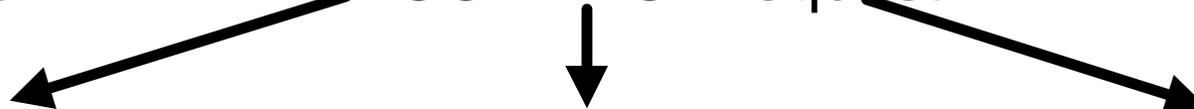


Nomes de primitivas

- Existem normas para a composição de nomes de primitivas.

- Exemplo:

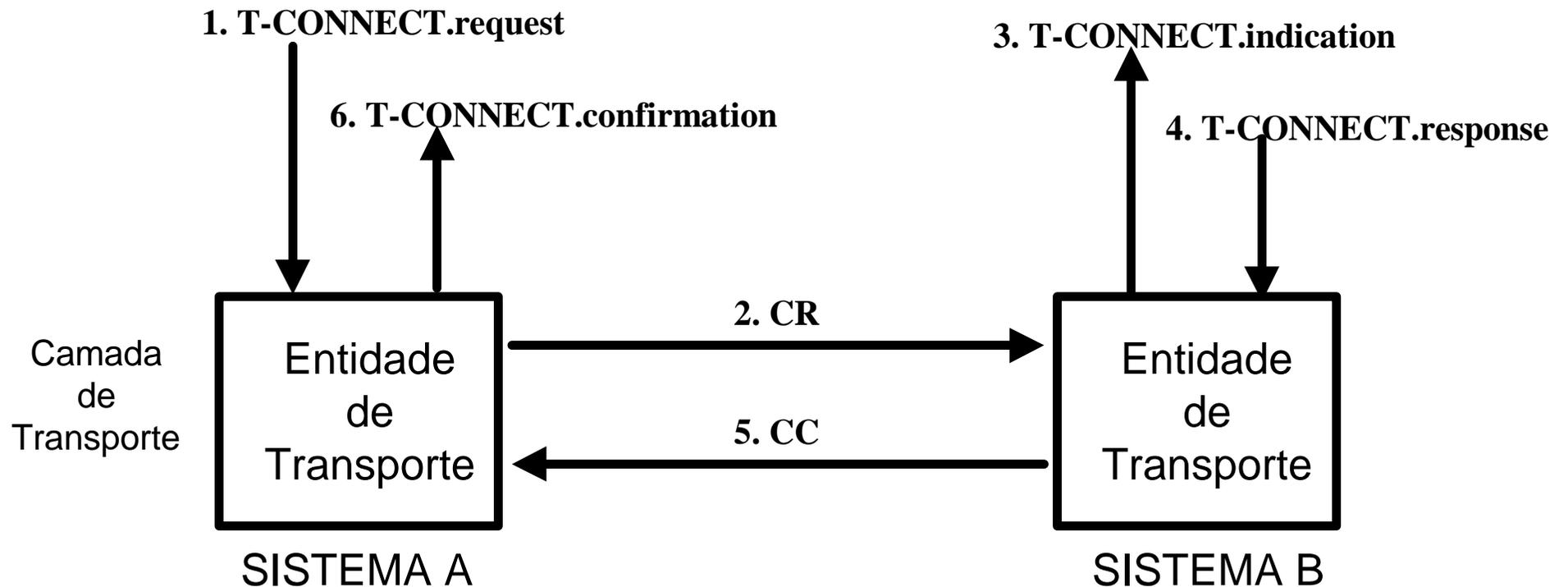
T-CONNECT.request



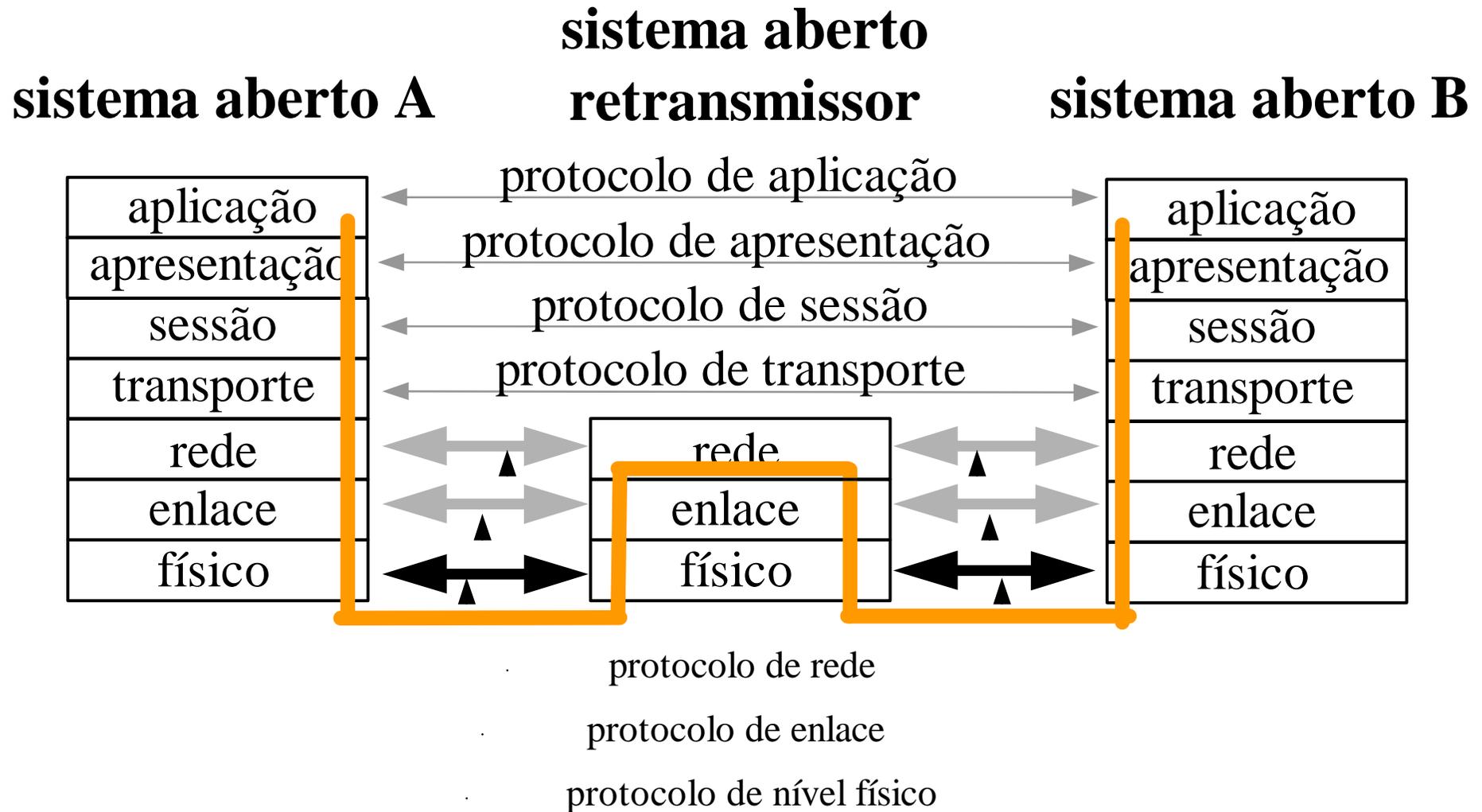
Iniciais da camada	Nome do serviço	Tipo de primitiva
6 - P (Presentation)	CONNECT	request
5 - S (Session)	DISCONNECT	indication
4 - T (Transport)	RELEASE	response
3 - N (Network)	DATA	confirmation
2 - DL (Data link)	EXPRESS_DATA	
1 - PH (Physical)	TOKEN-GIVE	
	...	
	etc	

Primitivas

- Exemplo: a primitiva `T-CONNECT.request` corresponde a um pedido de abertura de conexão entre duas entidades de transporte.



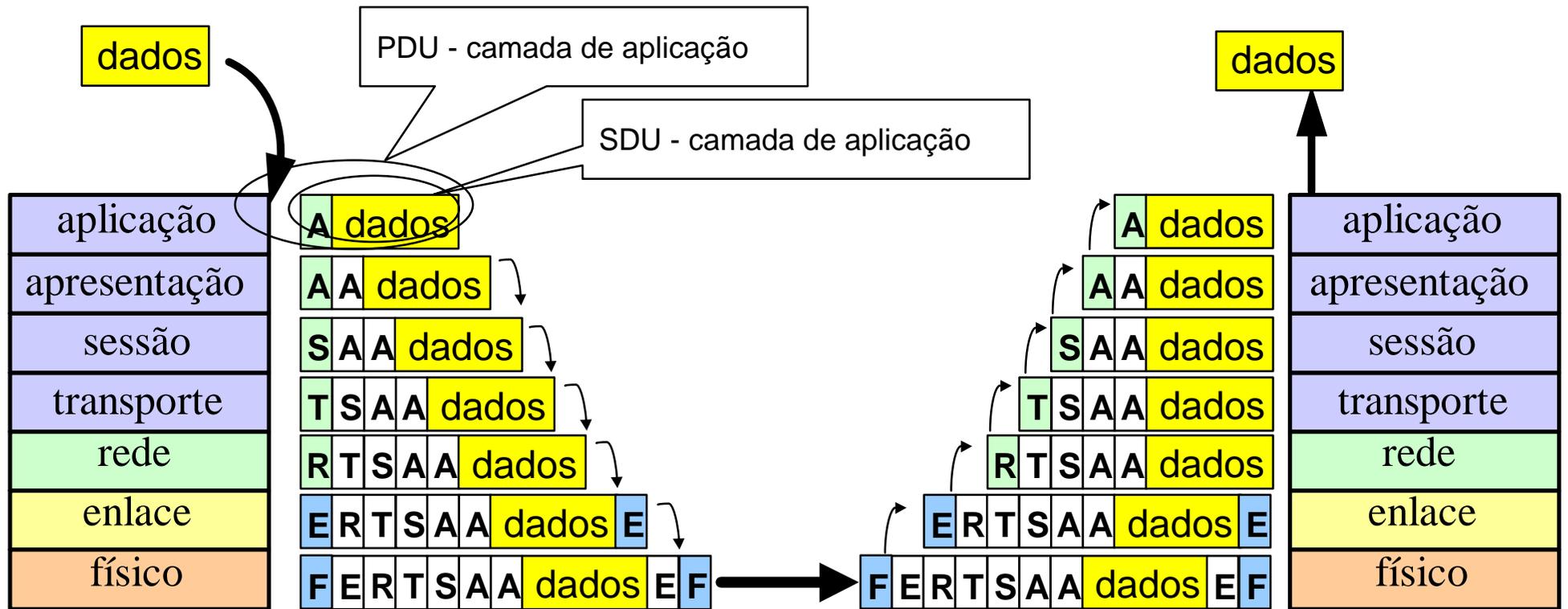
Níveis ou camadas do modelo OSI



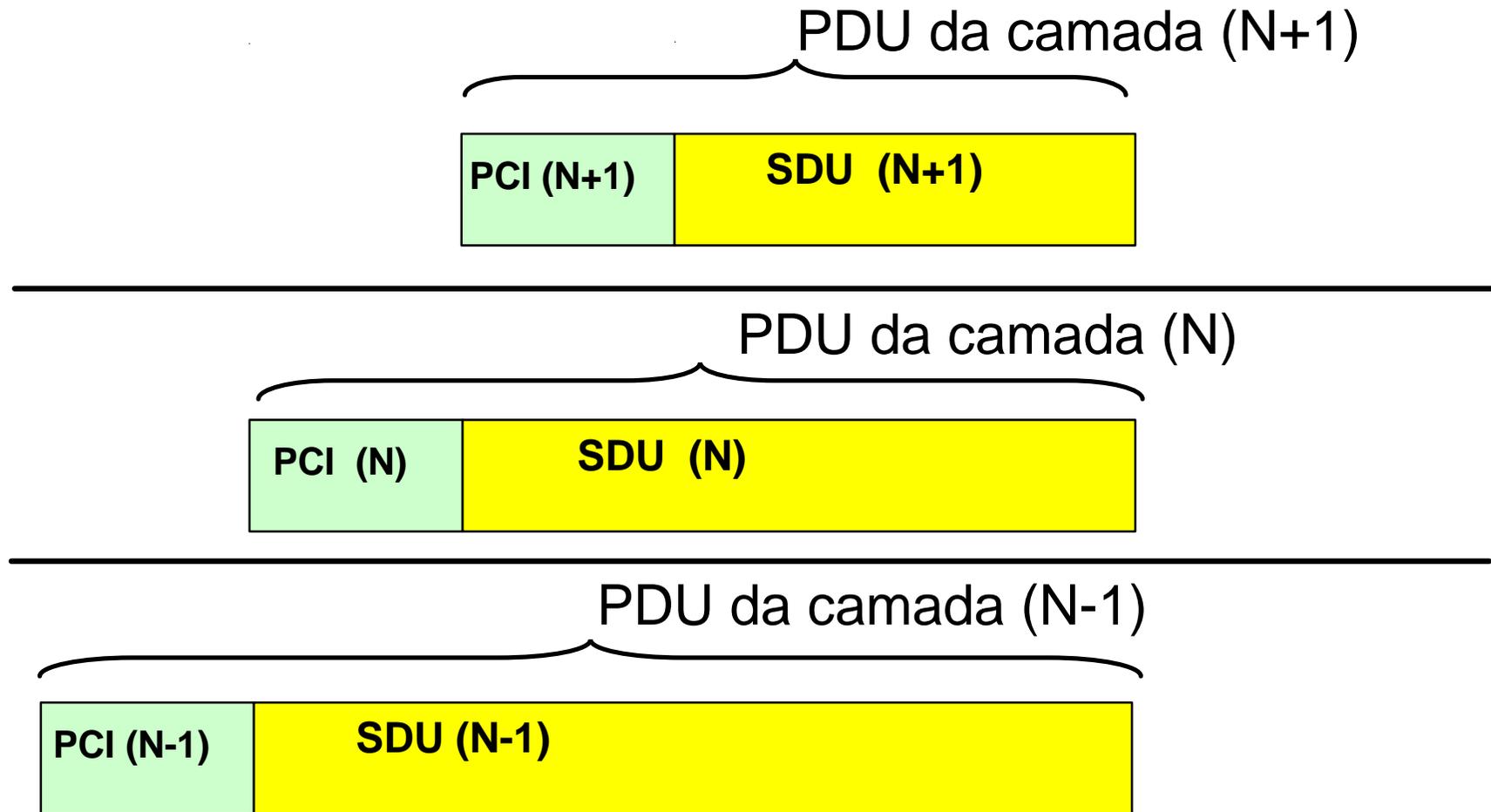
A troca de dados entre camadas

- Ao receber dados para efetuar um serviço, a camada N necessita incluir um cabeçalho, no qual são registradas informações relativas à camada.
- A esse cabeçalho, damos o nome de *Informação de Controle do Protocolo* - **PCI** (*Protocol Control Information*).
- Aos dados recebidos pela camada N, damos o nome de *Unidade de dados do Serviço* - **SDU** (*Service Data Unit*).
- Ao conjunto formado por PCI + SDU damos o nome de *Unidade de Dados do Protocolo* - **PDU** (*Protocol Data Unit*).
*Portanto, **PDU = PCI + SDU***

A troca de dados entre camadas



A troca de dados entre camadas



Camada ou nível físico

- A camada física trata do meio físico a ser utilizado: cobre, fibra ótica ou ondas de rádio.
- Nessa camada estão os padrões mecânicos, funcionais, elétricos e procedimentos para acesso a esse meio físico.
- Sua função principal envolve a transmissão transparente de seqüências de bits pelo meio físico, ou seja, sem se preocupar com seu significado, ou com a forma com que esses bits serão agrupados.
- Mantém a conexão física entre sistemas.
- Há vários tipos de conexão: Ponto-a-ponto ou multiponto, Full ou half duplex, Serial ou paralela.
- **NÃO É FUNÇÃO DESTA CAMADA SE PREOCUPAR COM ERROS DE TRANSMISSÃO**

Camada de Enlace

- A camada de **enlace de dados** (*data link layer*) esconde características físicas do meio de transmissão.
- Provê meio de transmissão confiável entre dois sistemas adjacentes.
- Tem a função de **detectar** e opcionalmente corrigir esses erros.
- Funções mais comuns:
 - Delimitação de quadro
 - Detecção de erros
 - Sequencialização
 - Controle de fluxo

Camada de Rede

- A camada de Rede (*network layer*) provê canal de comunicação independente do meio.
- Efetua operações de Funções características:
 - Acesso à sub-rede
 - Operação da rede
 - Interconexão de redes e de sub-redes
 - Chaveamento e roteamento \Rightarrow portanto, a camada superior (transporte) não precisa se preocupar sobre “como é que os dados são encaminhados até o destino”, que caminho fazem, etc.
- Normalmente operam em modo ***circuito virtual*** ou ***datagrama***
 - **Datagrama:** não possui conceito de conexão, cada pacote trafega independentemente dos demais pacotes que o antecedem/sucedem
 - **Circuito virtual:** é necessário que o transmissor envie um pacote especial, solicitando a abertura de conexão. Uma vez estabelecida a conexão (e a rota), os demais pacotes com o mesmo destino seguem o mesmo caminho.

Camada de Transporte

- A camada de transporte (***transport layer***) trata da transferência de dados transparente, isolando as camadas superiores dos detalhes de transmissão da rede e sub-rede.
- **Multiplexação** (união de várias conexões de transporte em uma conexão de rede) para permitir maior grau de compartilhamento de recursos; e **splitting** (uma conexão de transporte ligada a várias conexões de rede) para aumento de desempenho.
- Controles fim-a-fim:
 - controle de seqüência de pacotes
 - detecção e recuperação de erros de multiplexação
 - detecção e recuperação de erros básicos
 - controle de fluxo: cuida para que o transmissor não envie além do que o receptor consegue receber e tratar.

Camada de Sessão

- A camada cuida do **sincronismo de diálogo**
 - Recepção x transmissão
- Recupera conexões de transporte sem perder conexões de sessão.
- Possui mecanismos de verificação de sincronização.
- Não efetua multiplexação da camada de transporte.
- Utiliza mesma conexão de transporte para várias conexões de sessão não simultâneas.

Camada de Apresentação

- A camada de apresentação cuida da transparência de representação de dados: sintaxes abstratas.
- Faz transformações de dados, como : compressão de textos, conversões, criptografia, conversão de padrões de terminais e arquivos para padrões de rede, e vice-versa.
- Sintaxes:
 - Do transmissor
 - Do receptor
 - De transferência
- Contexto de apresentação:
Sintaxe abstrata + Sintaxe de transferência

Camada de Aplicação

- **A camada de aplicação desempenha funções específicas de utilização dos sistemas.**
- **Categoria de processos de aplicação**
 - **Correio eletrônico: X.400**
 - **Transferência de arquivos: FTAM**
 - **Serviço de diretório: X.500**
 - **Processamento de transações: TP**
 - **Terminal virtual: VT**
 - **Acesso a bancos de dados: RDA**
 - **Gerência de rede**
- **Formato de dados**
 - **EDI/EDIFACT: eletrônico data interch. for administration, commerce, and trading**
 - **ODA/ODIF: open document architecture/interchange format**
 - **CGM/CGMIF: computer graphics metafile/interchange format**