

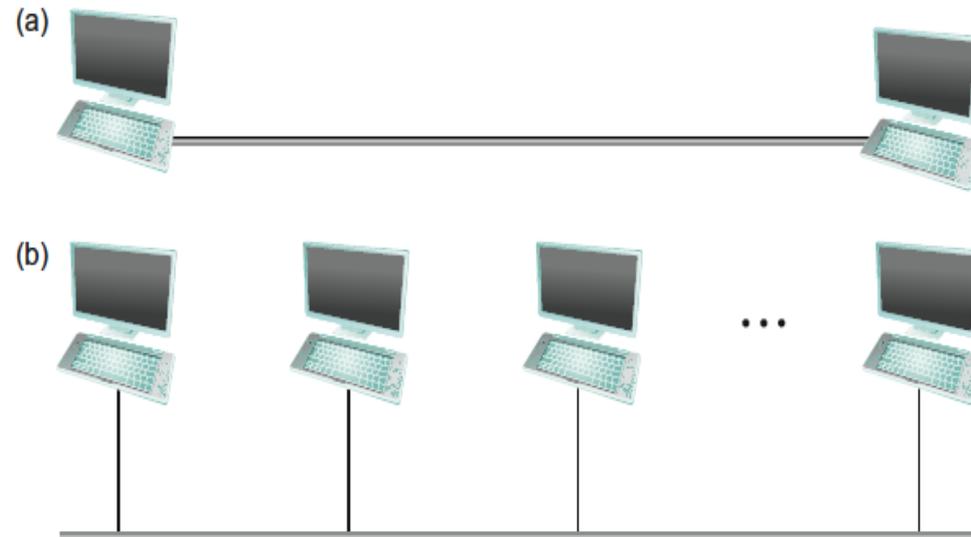


Internet

Modele arhitecturale de referință

Aplicatii Internet

- Internetul este cunoscut de utilizatori prin aplicatiile sale
 - e-Mail
 - Transferul de fisiere
 - Web
 - si multe altele
- Bazate pe **comunicarea** prin legaturi intre calculatoare
- **Legatura** poate fi directa, intre doua sau mai multe calculatoare (noduri)
 - **punct la punct** – prin fire
 - **acces multiplu** – de ex wireless (retele celulare sau Wi-Fi)



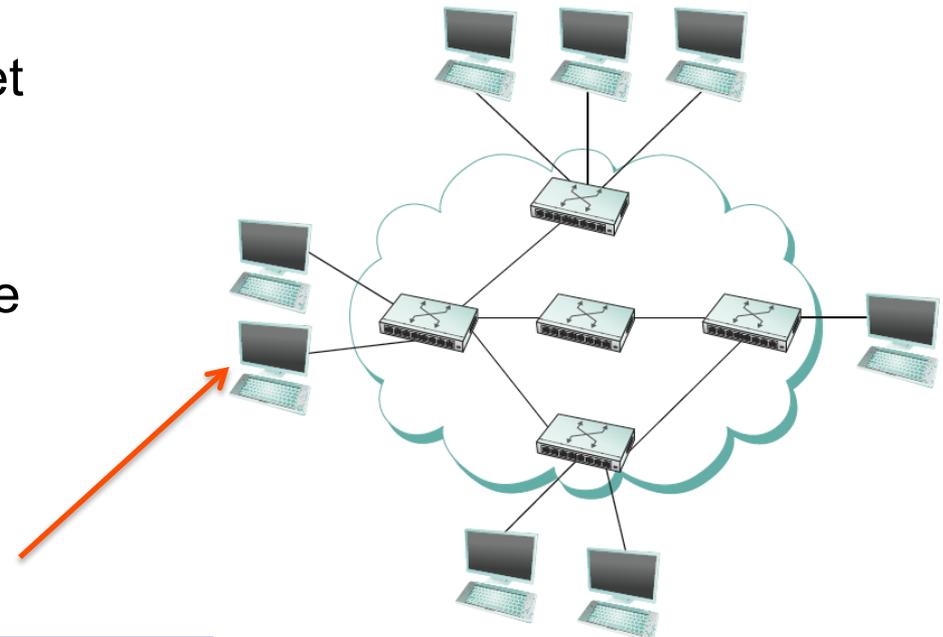


Legatura de date

- Legaturile asigura functionalitati specifice **datelor** (diferite de cele pentru comunicari vocale)
 - **codificarea** bitilor pentru transmisie si intelegerea codurilor la receptie
 - delimitarea sirurilor de biti (**incadrarea**) care constituie mesaje complete (**cadre**) ce pot fi livrate nodului receptor
 - **detectia** erorilor de transmisie si corectarea lor
 - controlul **fluxului** de date
 - **controlul accesului** la media (pentru acces multiplu)
- Legaturile directe intre noduri **nu ofera scalabilitate**

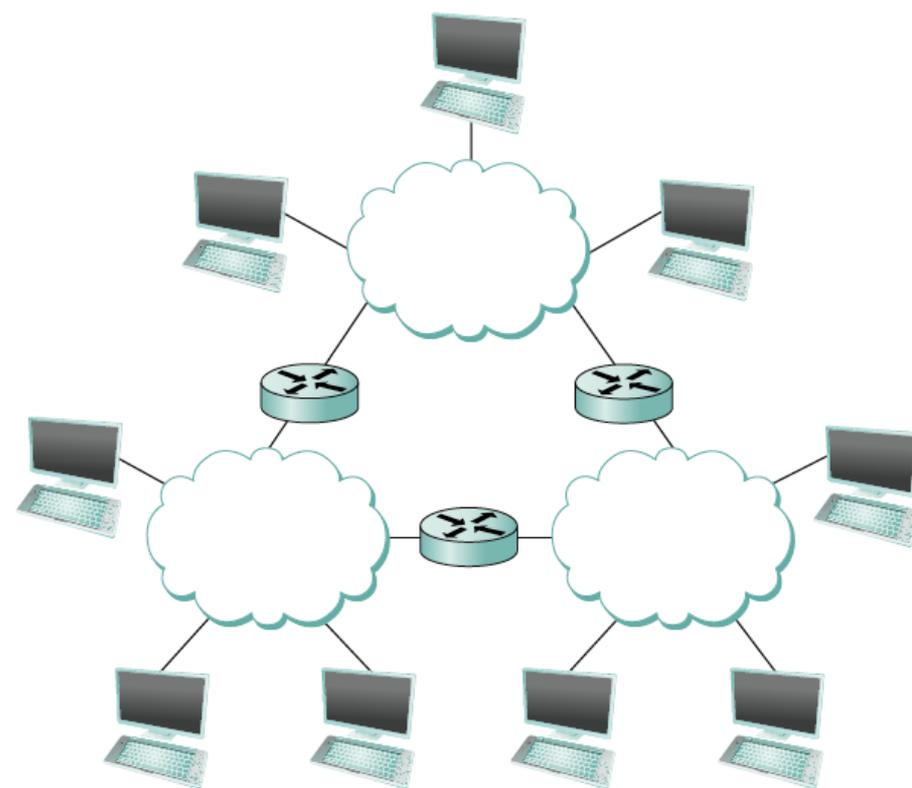
Retele

- Conexiunea intre doua noduri se poate face si **indirect**, prin **noduri intermediare** (switch-uri)
 - atasate la mai multe legaturi
 - au functii speciale
 - formeaza o retea
- Functioneaza dupa principiul sistemului **postal**; un nod
 - primeste cate un bloc complet de date (**pachet**)
 - il **memoreaza** temporar
 - il **re-transmite** (**dirijeaza**) catre destinatie, printr-una din legaturile la nodurile vecine
- Nodurile din afara retelei sunt numite **gazde** (ale aplicatiilor)
- Pentru ca retea sa functioneze, fiecarui nod i se ataseaza o **adresa** de retea pentru identificare
- Nodul **sursa** include in pachet **adresa** nodului **de destinatie**



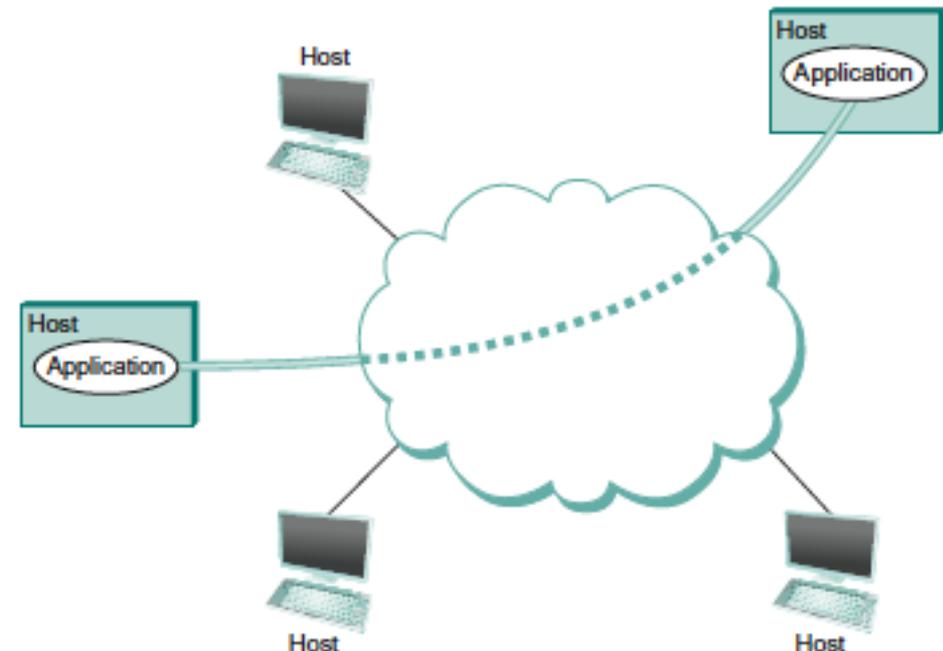
Retele interconectate

- Nodurile intermediare pot fi organizate în mai multe rețele interconectate
- Un nod conectat la mai multe rețele este numit **ruter** (sau **gateway**)
 - are funcții similare cu switch-ul
- Rețelele se bazează pe buna funcționare a **legăturilor** dintre noduri
- Rețelele transmit pachete între oricare două noduri neconectate direct
- Nu asigură **corectitudinea** transmisiei
 - e.g. pachete pierdute



Comunicarea între aplicații

- Reteaua trebuie să asigure comunicarea **între aplicații** din calculatoare diferite, adică să ofere **canale "logice"** prin care **procesele** de aplicație să comunice între ele
- Aceasta presupune facilități suplimentare peste cele de rețea
- **identificarea** unică a fiecărui **capăt de canal logic** printr-o pereche (adresa de rețea, **port**)
- mecanisme de **trimitere / primire** de mesaje de către procese
- garantarea **recepției corecte** a mesajelor
- livrarea mesajelor în **ordinea** în care au fost transmise
- pastrarea **confidențialității**
- etc.





Cerintele aplicațiilor

- Modul în care este folosit un canal logic diferă de la o aplicație la alta
- În **Web**, comunicarea se face între două procese
 - la comanda unui utilizator, un proces **client** (browser) trimite o **cerere** către un **server** Web – un mesaj care include identificatorul paginii dorite
 - procesul server trimite un răspuns către client – un mesaj care conține pagina Web solicitată, pe care clientul o afișează pe ecran
- În aplicații de livrare de conținut **audio/video**
 - transmiterea cererii este similară cu Web
 - livrarea este diferită
 - se trimite ca răspuns o serie de mesaje
 - conținutul este livrat utilizatorului pe măsura ce mesajele sunt primite
 - impune respectarea unor constrângeri de timp

Arhitectura

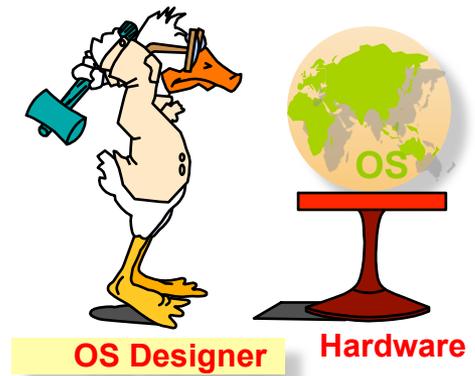
Arhitecture este arta si stiinta proiectarii cladirilor si a altor structuri fizice.



Acopera integrarea cladirilor cu mediul inconjurator, precum si detalii de constructie (de ex. mobila).

Se extinde la arta si disciplina crearii unui **plan** pentru un obiect sau sistem complex.

Usureaza intelegerea si realizarea sistemelor.





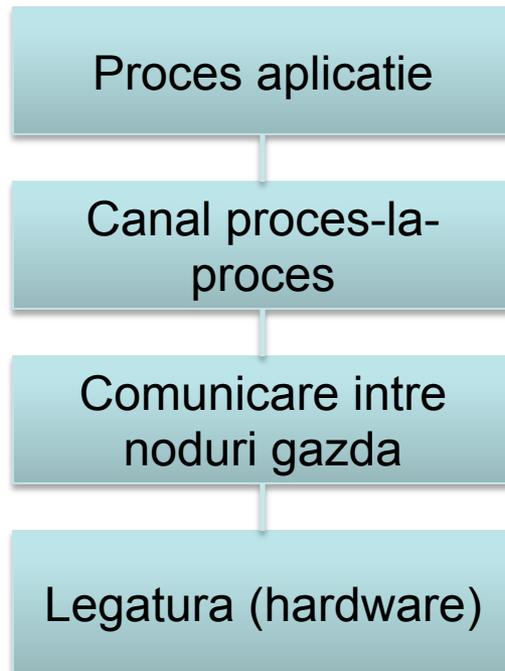
Arhitectura Retelelor de Calculatoare

- Retelele de calculatoare sunt sisteme complexe
- Abordarea lor ca **ansambluri de componente** simplifica intelegerea si realizarea
- **Arhitectura** descrie
 - modul in care **componentele** sunt organizate si
 - felul in care ele **interactioneaza**
- O astfel de descriere se numeste **model de referinta**
- Fiecare componenta
 - furnizeaza un **serviciu** util celorlalte componente
 - serviciul este furnizat printr-o **interfata** accesibila celorlalte componente
 - detaliile de implementare sunt ascunse altor componente

Arhitectura stratificata

In cazul retelelor arhitectura este **stratificata** (structurata pe mai multe **nivele**)

- serviciul oferit de un nivel foloseste serviciile nivelelor inferioare
- exemplul ilustreaza nivelele pentru serviciile discutate anterior
- exista **arhitecturi standard** pe care le discutam in continuare

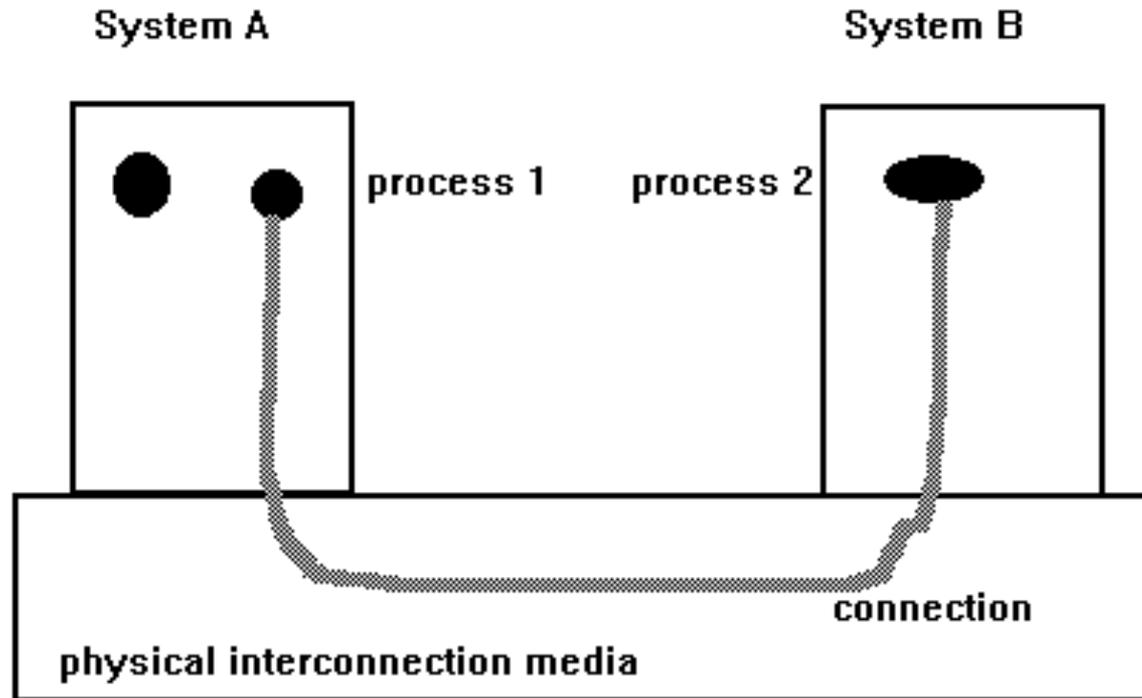


In figura se arata componentele dintr-un nod gazda si felul in care sunt interconectate. Fiecare **componenta** apartine unui **nivel** diferit.

Pot fi mai multe componente pe un nivel. Structuri similare se gasesc in celelalte noduri gazda.

Componente de bază

- Componente de baza ale unui model de comunicare:

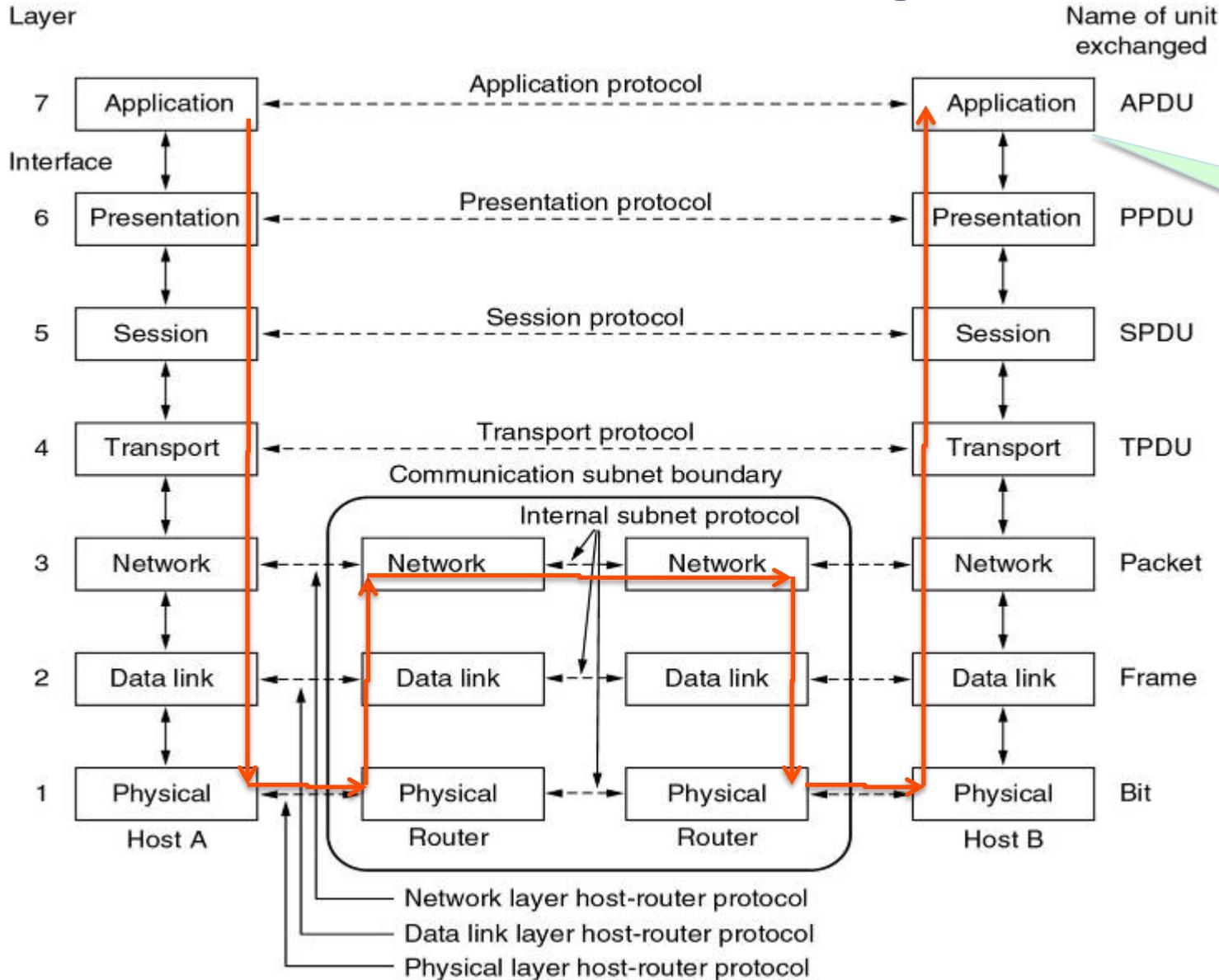


Protocol – set de reguli respectate de parti pentru a comunica intre ele; se refera la

- **formatul** mesajelor communicate: continut + meta-descriere (antet)
- **operatiile** executate
 - trimite cerere conectare, primeste raspuns, confirma raspuns etc.



Modelul de Referință ISO OSI



entitate de protocol

- Fiecare entitate are legaturi
- cu alte entitati din acelasi nod (prin interfețe de serviciu) și
 - cu entitati pereche din alte noduri (prin mesaje)



Nivel fizic

- **Funcție** - transmitere a șirurilor de biți pe un canal de comunicație
- Principalele **probleme**
 - codificarea zerourilor și a unităților
 - stabilirea și desființarea conexiunilor fizice
 - modul de transmisie (semiduplex sau duplex) etc.
- **Exemplu**
 - 802.11 Wi-Fi

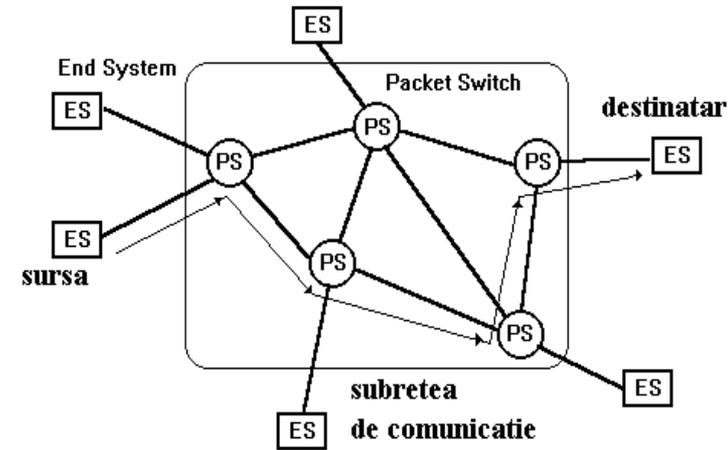
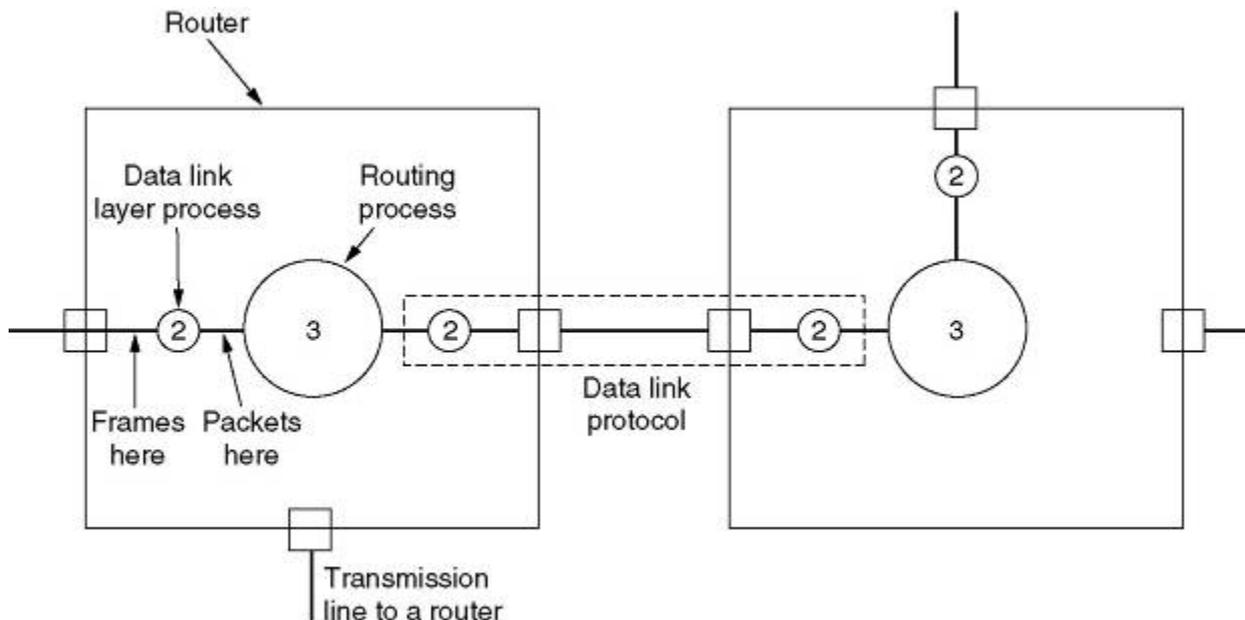


Legătura de date

- **Funcție** - realizează **agregarea** unor siruri de biti in **cadre** si **comunicarea lor sigură** și **eficientă** între două noduri adiacente (conectate printr-un canal fizic de comunicație)
- **Probleme rezolvate**
 - Încadrare
 - Control erori
 - Control flux
 - Transmisie transparentă
 - Management legătură
- **Exemplu** de protocol: HDLC (High Level Data Link Control)
flag address command data FCS flag
- **Implementare** prin
 - adaptoare de retea
 - drivere din sistemul de operare al calculatorului

Nivel rețea

- **Funcție** - transmiterea pachetelor între oricare două noduri din rețea
 - prin dirijarea lor de la un nod la altul
- **Probleme rezolvate**
 - alegerea legăturii următoare (dirijarea)
 - adresarea
 - calculul tabelului de dirijare



Nivel Transport

Funcție - asigură un transfer de date corect, eficient **între procese** din sistemul sursă și din sistemul destinatar

Oferă

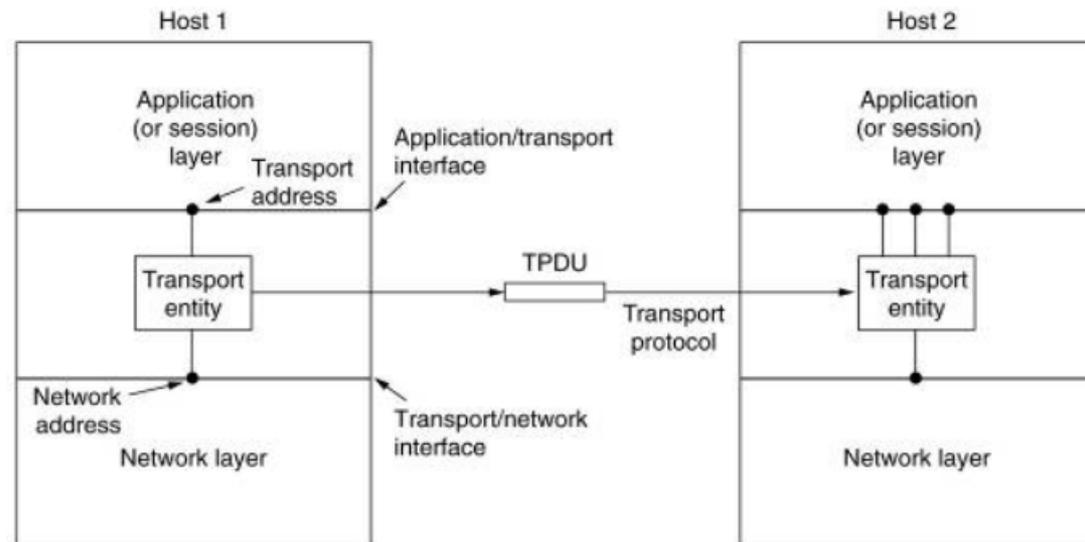
- un transfer sigur al datelor, chiar cu o rețea nesigură;
- o **interfață uniformă** pentru nivelul superior, independent de tipul rețelei utilizate.

Separă două categorii de nivele

- furnizorul serviciilor de transport (nivele 1-4)
- utilizatorul serviciilor de transport (nivele 5-7)

Probleme

- gestiunea conexiunilor
- transferul datelor
- controlul fluxului
- adresarea





Nivele superioare

- **Nivel Sesiune**
 - Controlul dialogului între aplicații
 - Sincronizarea transferurilor
 - Stabilirea unor puncte de verificare și reluare a transferurilor
- **Nivel Prezentare**
 - Conversia formatului datelor între
 - sintaxa folosită de aplicații și
 - sintaxa de transfer



Nivel Aplicație

- Servicii comune unor categorii de aplicații
 - Mesagerie
 - Transfer de Fișiere
 - Terminal Virtual
 - Serviciu de Directoare

Ierarhii de protocoale

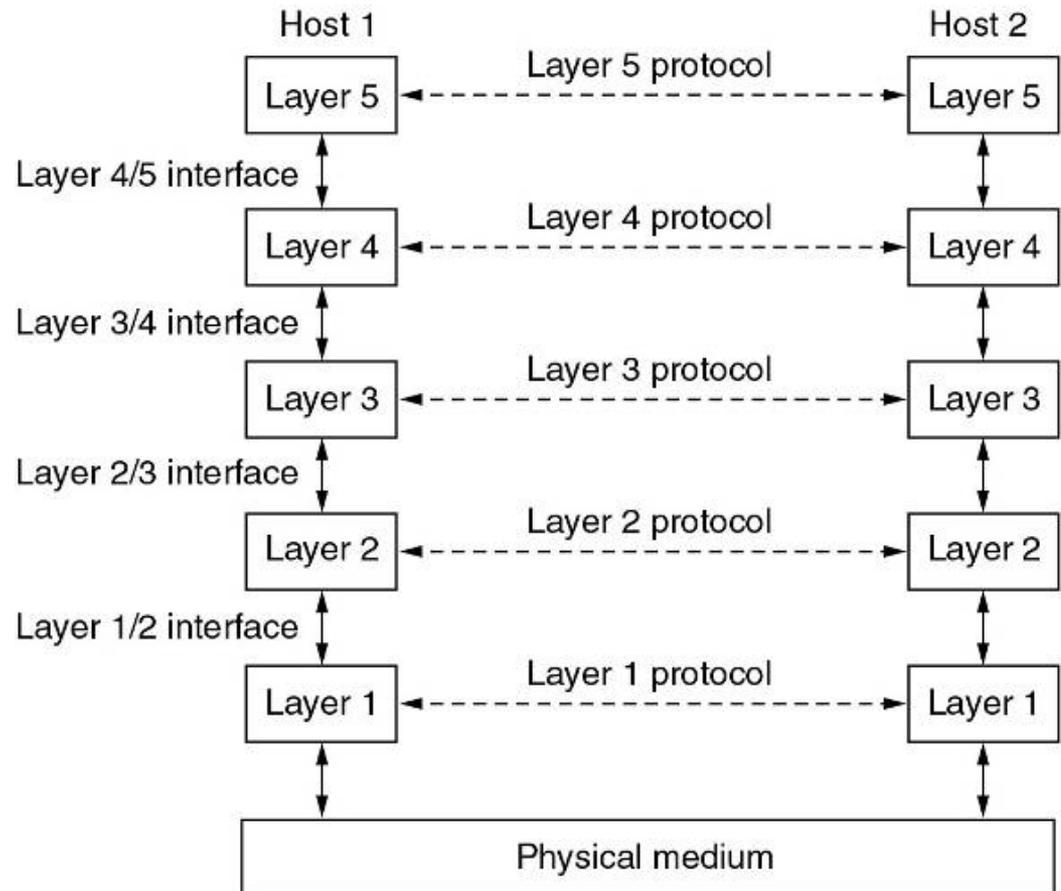
Complexitatea **conexiunii** → organizarea pe mai multe **nivele** cu functii distincte

- **arhitectura rețelei** = setul de nivele si protocoale
- **stiva de protocoale** = lista ordonata a protocoalelor folosite

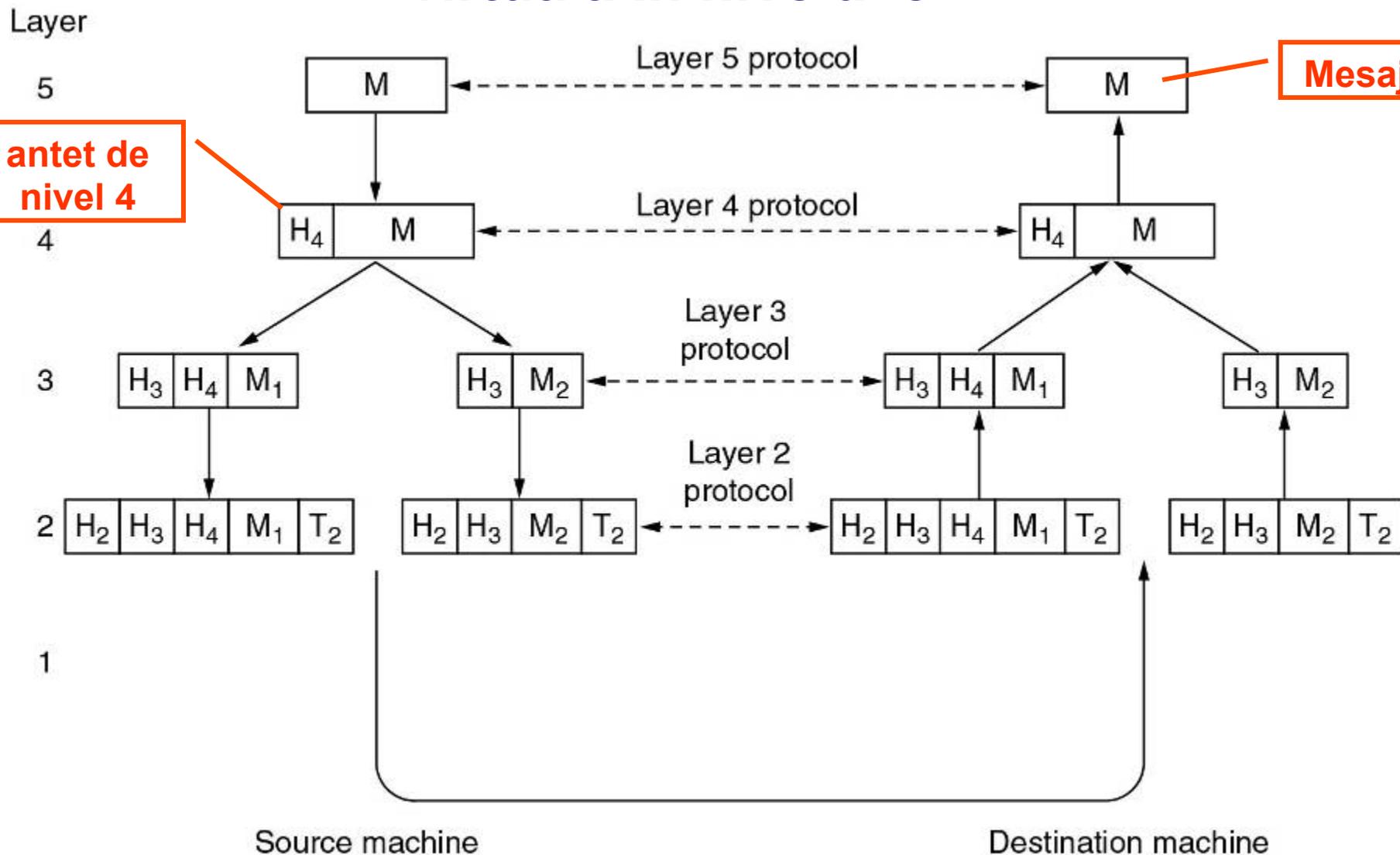
Interfata defineste **serviciul (operatiile primitive)** pe care un nivel il ofera nivelului de deasupra

Ex. de primitive:

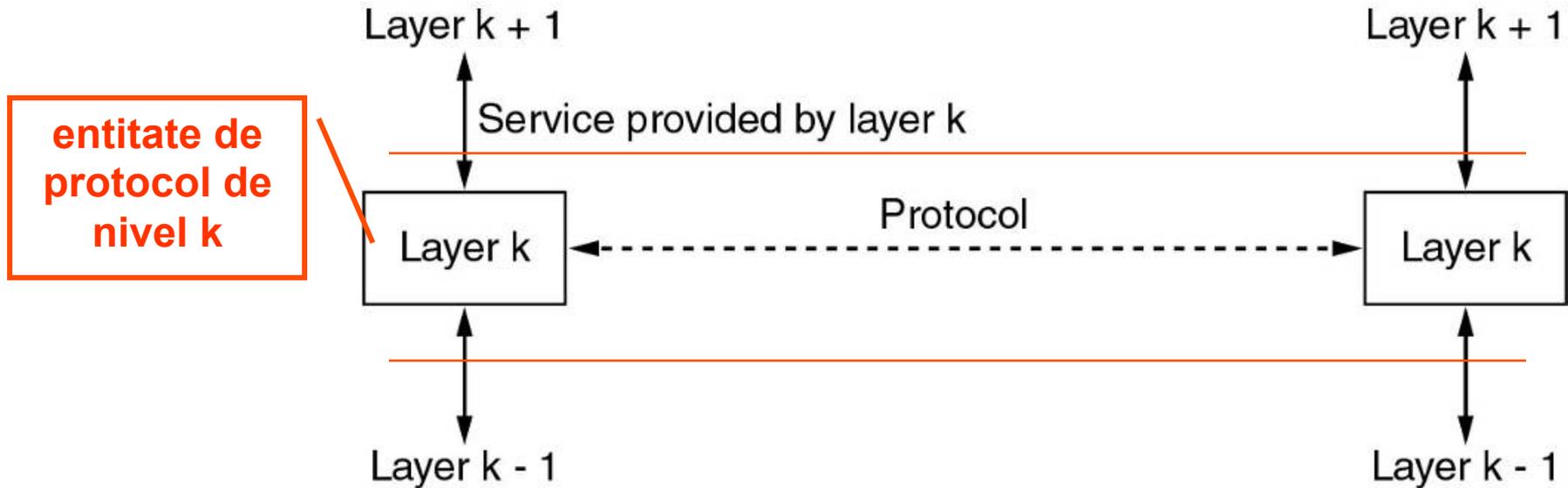
- connect
- accept
- send
- receive
- disconnect



Flux de informație suportând o comunicare virtuală în nivelul 5



Relația între servicii și protocoale



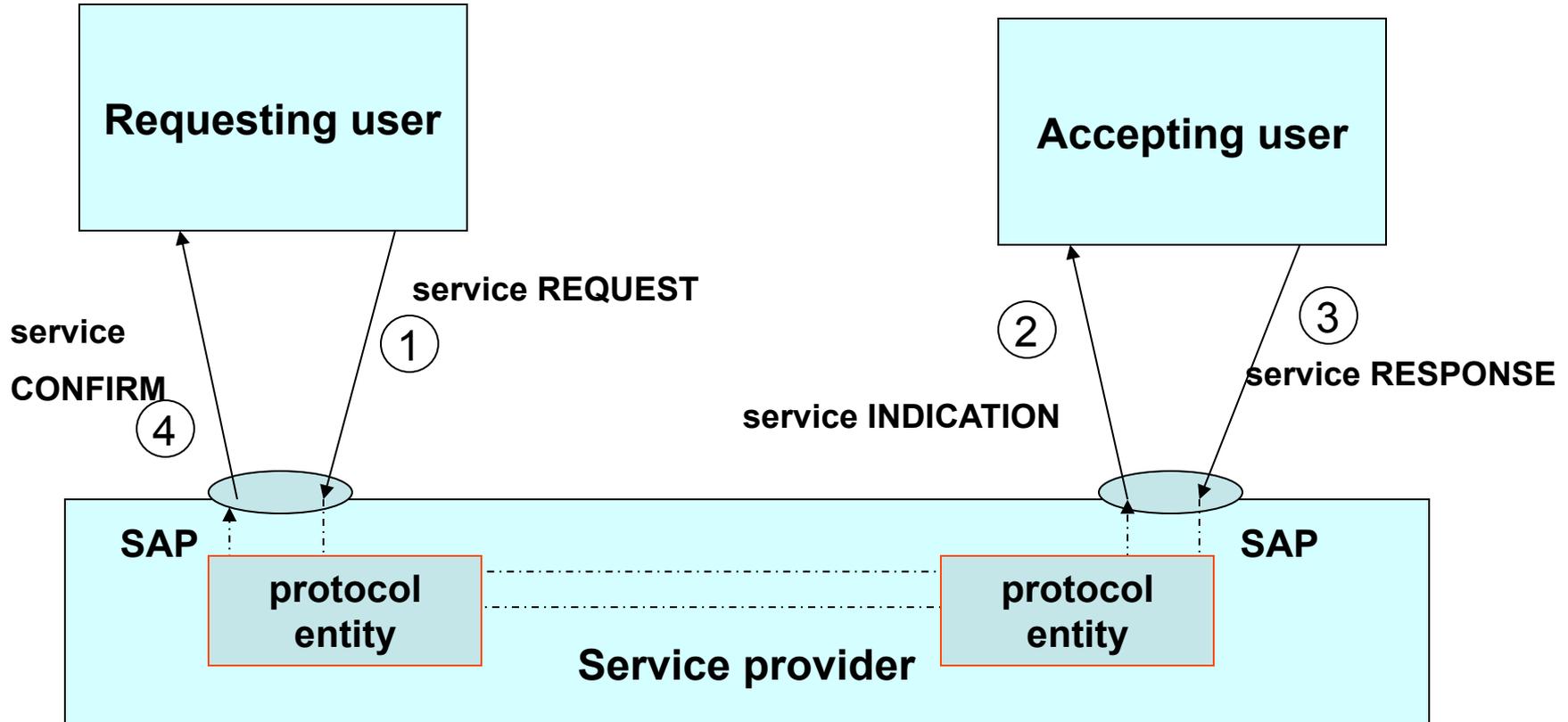
- entitățile de protocol din nivelul k
- colaborează folosind **protocolul** de nivel k
- folosind **serviciul** nivelului k-1
- pentru a furniza **serviciul** de nivel k
- **entitatilor** aflate pe nivelului k+1



Primitive de serviciu

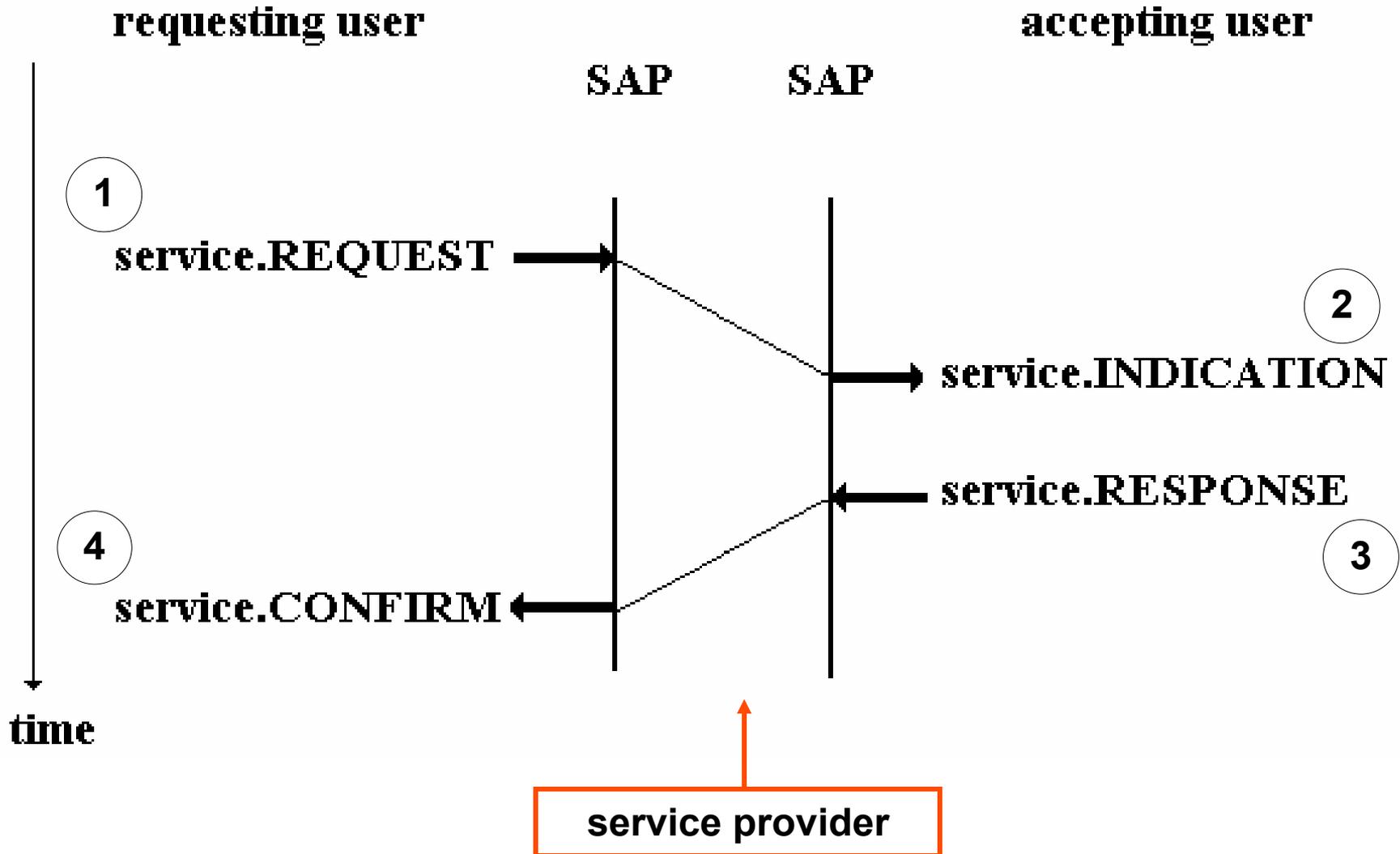
- Un serviciu este specificat de un set de **primitive** (operații accesibile utilizatorului serviciului)
- Patru clase de primitive
 - REQUEST cere un serviciu
 - INDICATION anunța producerea unui eveniment
 - RESPONSE răspuns la eveniment
 - CONFIRM confirmă cererea

Servicii confirmate

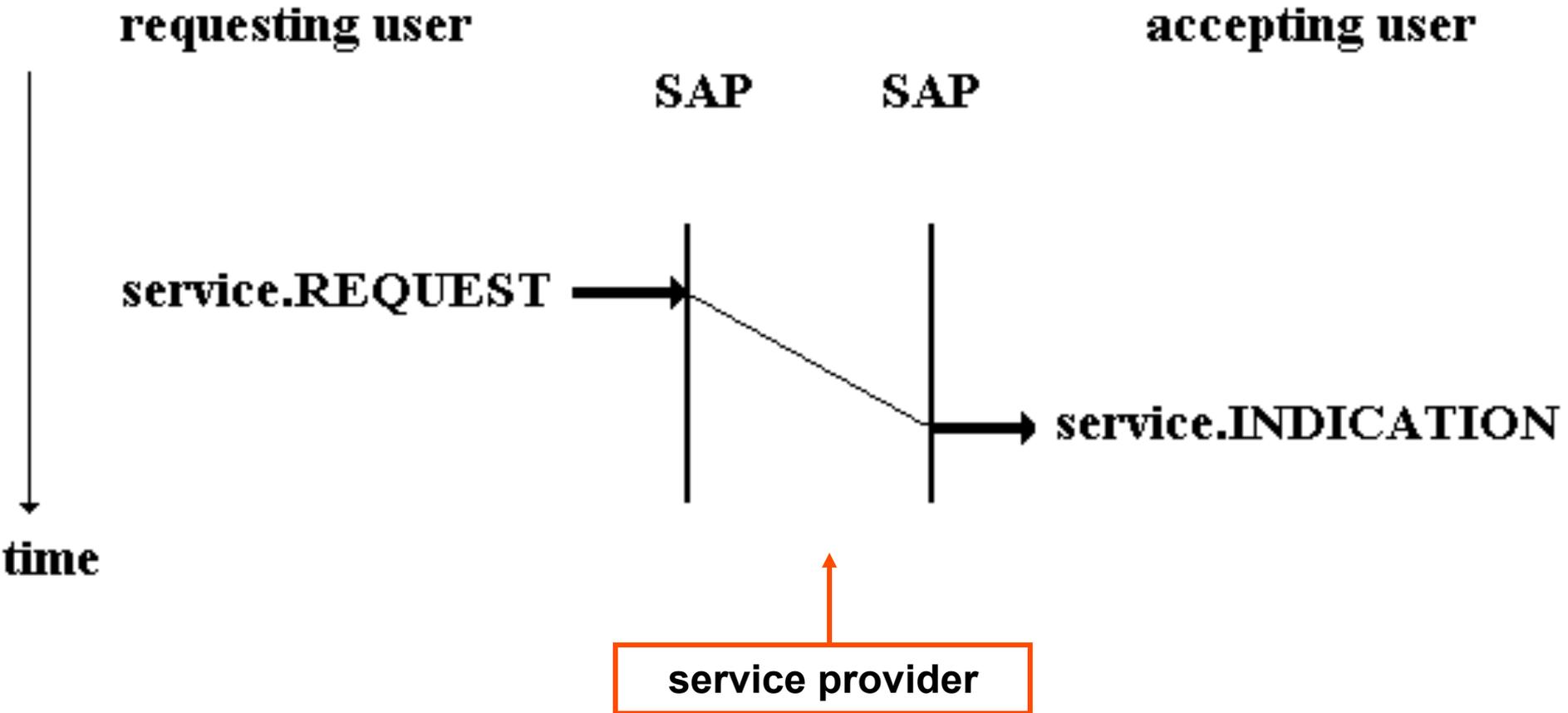


- Entitățile de protocol pot inter-schimba mai multe mesaje pentru un singur serviciu confirmat
 - ex.: mesaje de negociere a serviciului, de repetare la eroare etc.

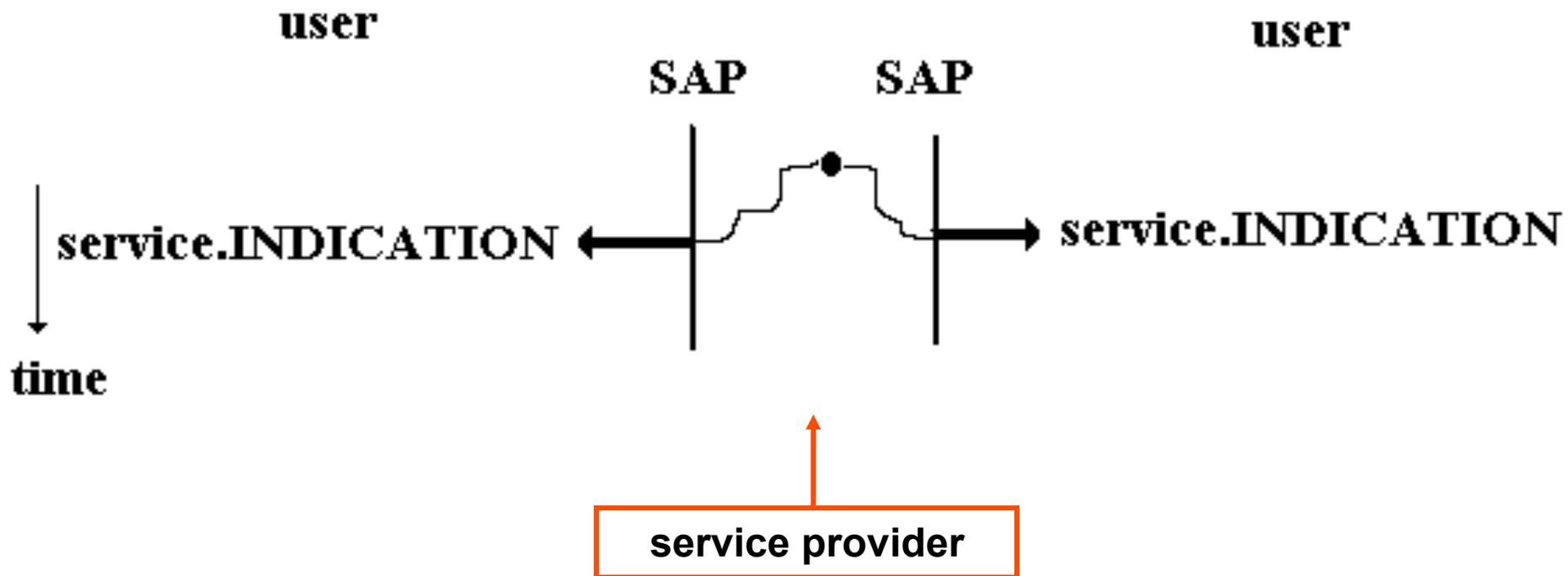
Servicii confirmate (o alta reprezentare)



Servicii neconfirmate



Servicii inițiate de furnizor





Mod orientat pe conexiune

utilizator solicitant

utilizator solicitat

connect.request	----->		--->	connect.indication
connect.confirm	<----	furnizor	<---	connect.response
data.request	----->		--->	data.indication
data.request	----->	serviciu	--->	data.indication
disconnect.request	----->		--->	disconnect.indication



Ce conțin specificațiile ?

Specificație Serviciu

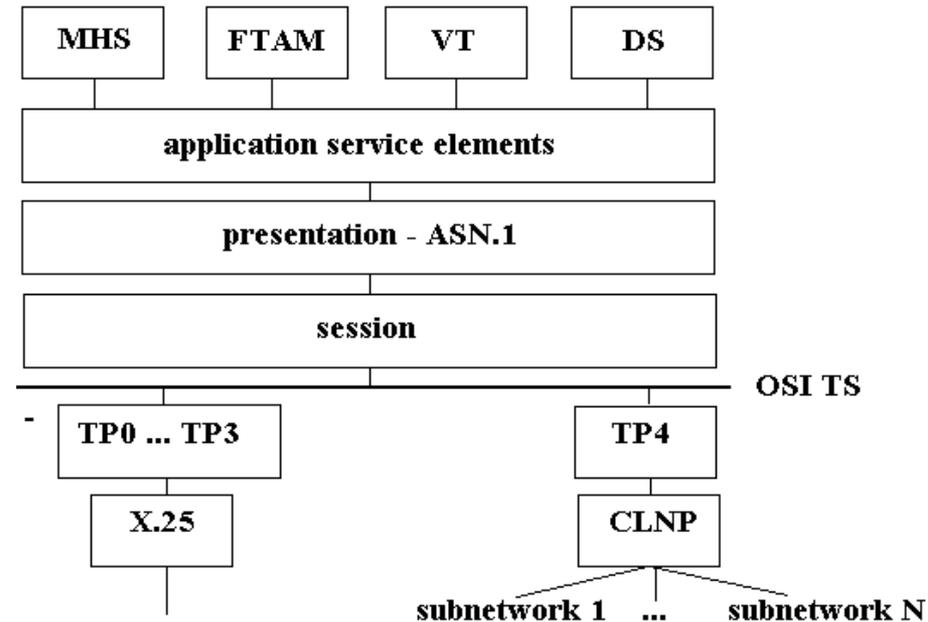
- primitive (operații)
- parametri
- reguli asupra ordinii operațiilor (state machine)

Specificație Protocol

- scop și funcții
- servicii oferite
- servicii utilizate din nivel inferior
- structura internă (entități și relații)
- tipuri și formate mesaje schimbate între entități
- reguli de reacție a fiecărei entități la comenzi, mesaje și evenimente interne

Protocole OSI

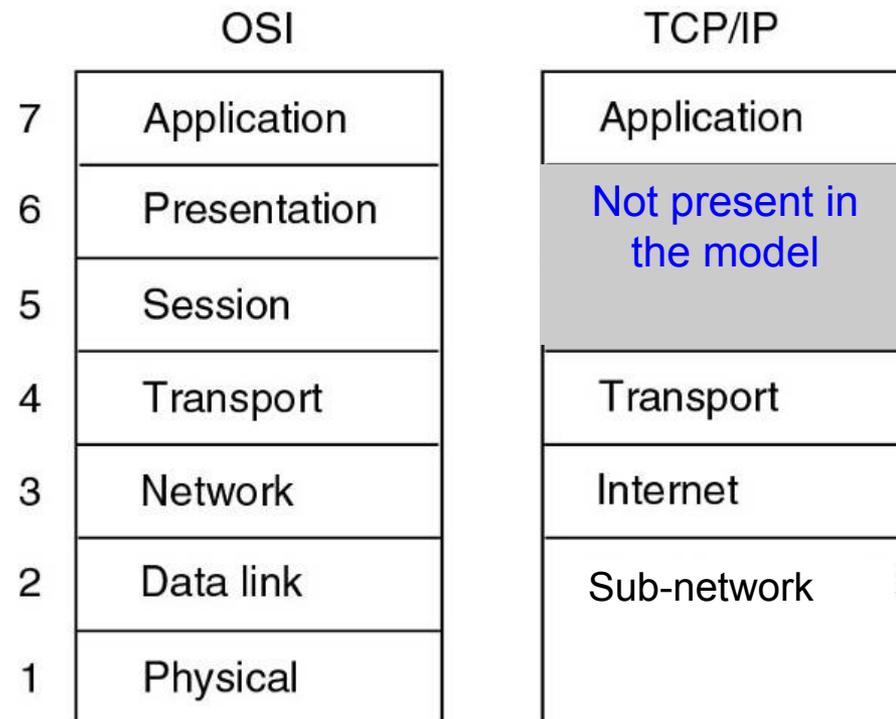
Physical layer:	V10, V11, V24, V35 X.21, EIA RS-232-D MAC for LANs ISDN physical interface
Data Link Layer:	HDLC LAP B for X.25 LLC for LAN LAP D for ISDN
Network Layer:	X.25, X.3, X.28, X.29 CLNP
Transport Layer:	TP0,..., TP4
Session Layer:	session protocol
Presentation Layer:	ASN.1
Application Layer:	MHS - Message Handling System, X.400 FTAM - File Transfer, Access, and Management VT - Virtual Terminal DS - Directory Services, X.500





Modelul de referință TCP/IP

- Nivelul inferior este **sub-retea**
 - mare varietate de protocoale (ex. 802.11 Wi-Fi)
 - pot fi mai multe sub-nivele
 - modelul nu da detalii despre acest nivel
- Nivelul Internet
 - un singur protocol, IP
 - pentru rețele interconectate
- Nivelul Transport
 - TCP – canal sigur pentru siruri de octeti
 - UDP – canal nesigur pentru livrarea datagramelor (user datagram – sinonim pentru mesaje)





Protocoale în modelul TCP/IP

- Nivel Aplicatie
 - Varietate de protocoale pentru transferul fișierelor și poștei, login la distanță, managementul rețelei

FTP	SMTP	Telnet	DNS	SNMP		
TCP			UDP		ICMP	EGP
	ARP	IP				
Ethernet	LLC 802.2			X.25	Packet Radio	
	MAC 802.3	MAC 802.4	MAC 802.5			
	Ethernet 802.3	Token Bus 802.4	TokenRing 802.5			



Alte protocoale în modelul TCP

HTTP	HyperText Transfer Protocol
IIOp	Internet Inter-Orb Protocol
WAP	Wireless Application Protocol
SOAP	Simple Object Access Protocol
LDAP	Lightweight Directory Access Protocol
SSL	Secure Sockets Layer
VPN	Virtual Private Networks
IPSEC	IP Security
PKI	Public Key Infrastructure
HTML	HyperText Markup Language
XML	Extensible Markup Language
WSDL	Web Services Description Language
UDDI	Universal Description, Discovery, and Integration



Comparație OSI și TCP/IP

Contra OSI

- Moment nepotrivit
- Tehnologie proastă
- Implementări rele
- Politici proaste

Contra TCP-IP

- Nu distinge între servicii, interfețe, protocoale
- Nu este un model general
- “Nivelul” gazdă-rețea nu este un nivel
- Nu menționează nivelele fizic și legătură de date
- Protocoale minore bine înrădăcinate - greu de înlocuit