

Distributed Systems



Alessio Vecchio

alessio.vecchio@unipi.it

Dip. di Ingegneria dell'Informazione
Università di Pisa

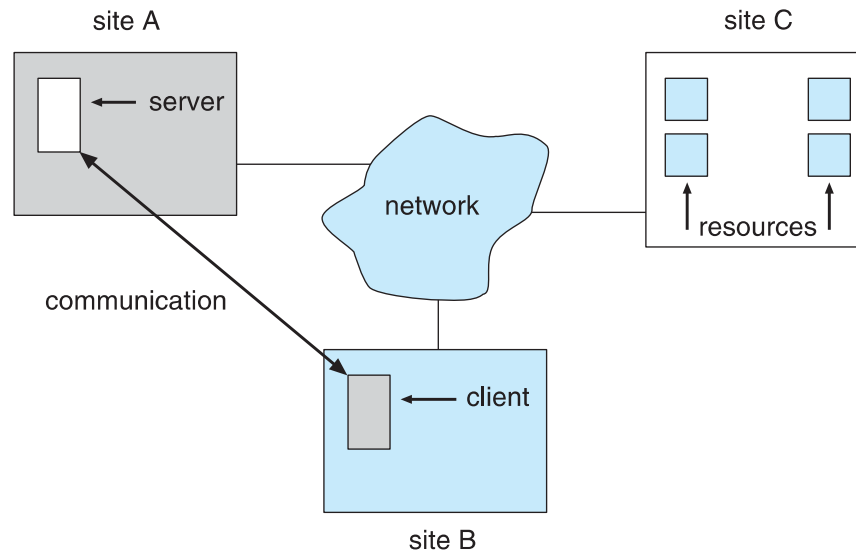
Based on original slides by Silberschatz, Galvin, and Gagne
Operating System Concept

Distributed Systems

- Introduction
- Communication Networks
- Communication Protocols
- Distributed Programming

Overview

- A **distributed system** is a collection of loosely coupled nodes interconnected by a communications network
- Nodes variously called **processors, computers, machines, hosts**
 - **Site** is location of the machine, **node** refers to specific system
 - Generally a **server** has a resource a **client** node at a different site wants to use



Reasons for Distributed Systems

- **Resource sharing**
 - Sharing files or printing at remote sites
 - Processing information in a distributed database
 - Using remote specialized hardware devices such as ***graphics processing units*** (GPUs)
- **Computation speedup**
 - Distribute subcomputations among various sites to run concurrently
 - **Load balancing** – moving jobs to more lightly-loaded sites
- **Reliability**
 - Detect and recover from site failure, function transfer, reintegrate failed site

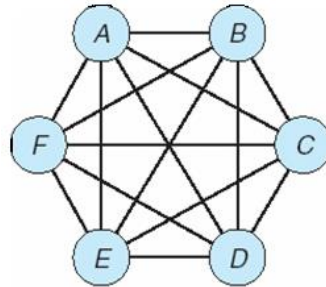
Network Operating Systems

- Users are aware of multiplicity of machines
- Access to resources of various machines is done explicitly by:
 - Remote logging into the appropriate remote machine (ssh)
 - `ssh dii.unipi.it`
 - Transferring data from remote machines to local machines, via the File Transfer Protocol (FTP) mechanism
 - Upload, download, access, or share files through cloud storage
- Users must change paradigms – establish a **session**, give network-based commands, use a web browser
 - More difficult for users

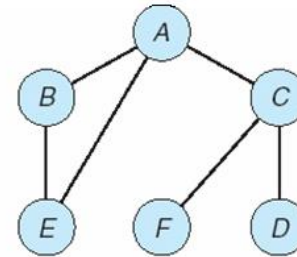
Connection Strategies

- **Circuit switching** - A permanent physical link is established for the duration of the communication (i.e., telephone system)
- **Message switching** - A temporary link is established for the duration of one message transfer (i.e., post-office mailing system)
- **Packet switching** - Messages of variable length are divided into fixed-length packets which are sent to the destination
 - Each packet may take a different path through the network
 - The packets must be reassembled into messages as they arrive
- Circuit switching requires setup time, but incurs less overhead for shipping each message, and may waste network bandwidth
 - Message and packet switching require less setup time, but incur more overhead per message

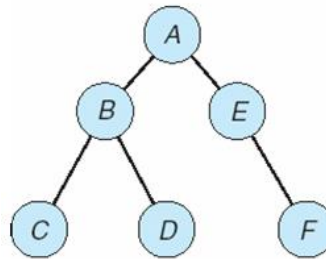
Network Topologies



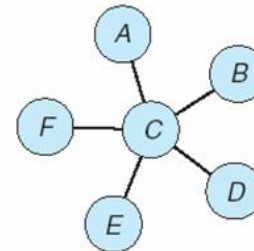
fully connected network



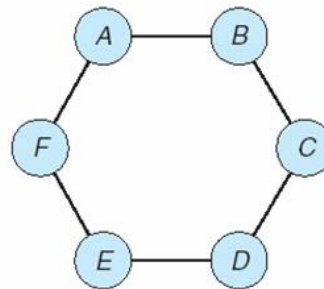
partially connected network



tree-structured network



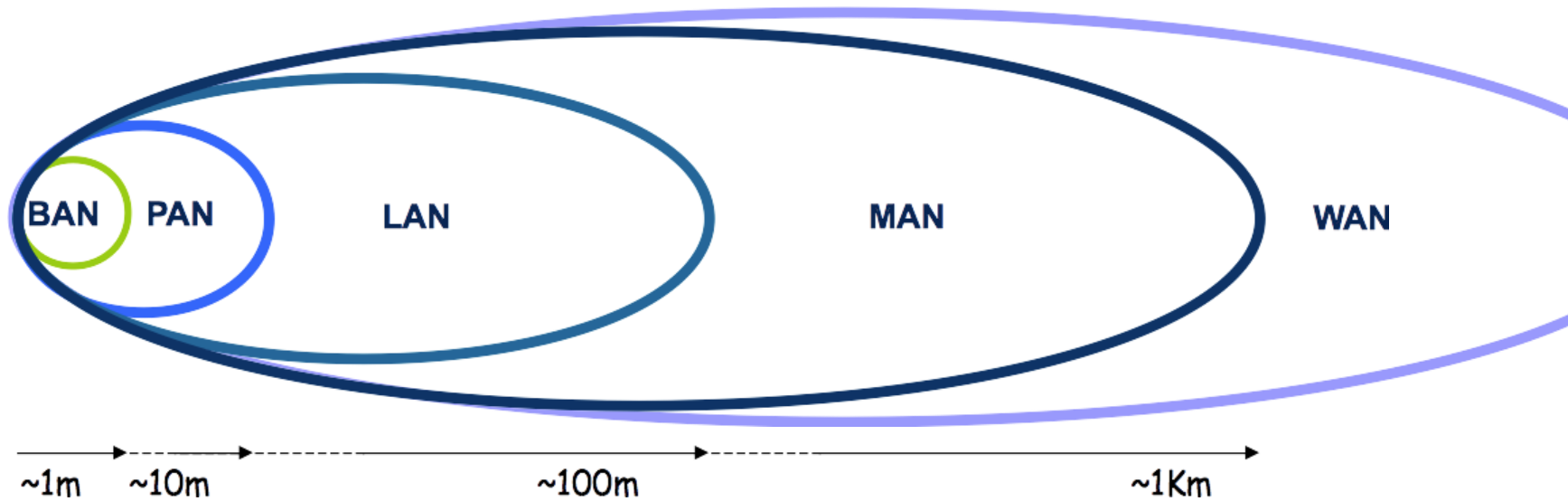
star network



ring network

Classification of Communication Networks

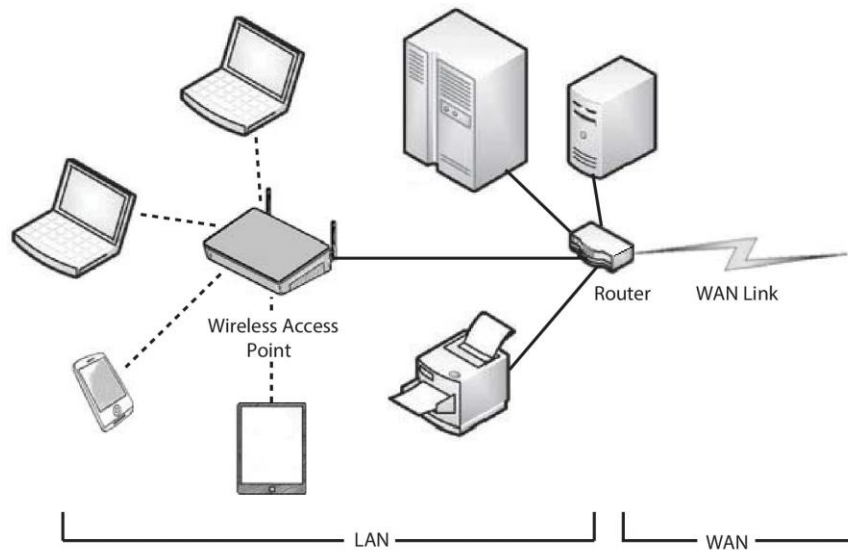
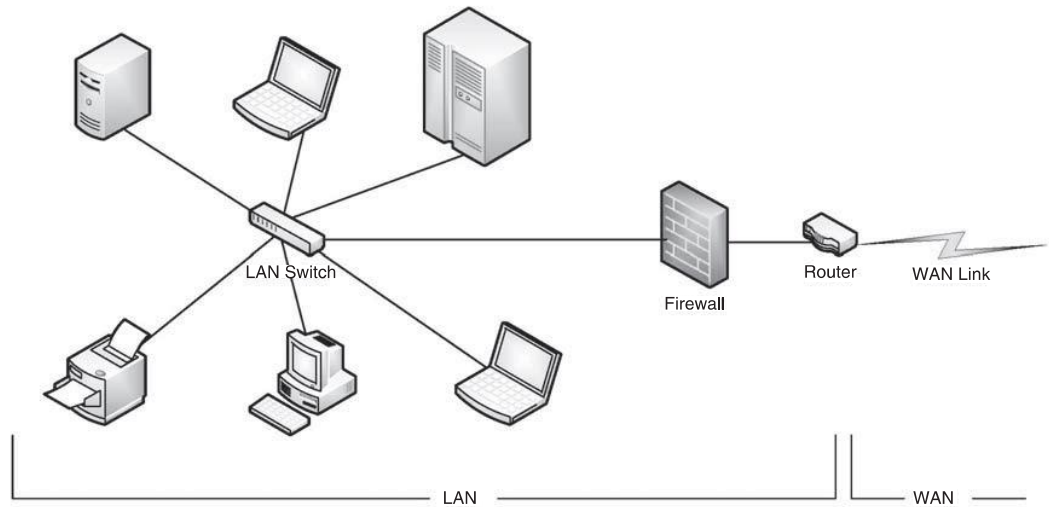
- Wide Area Networks (WAN)
- Metropolitan Area Networks (MAN)
- Local Area Networks (LAN)
- Personal Area Networks (PAN)
- Body Area Networks (BAN)



Network Structure

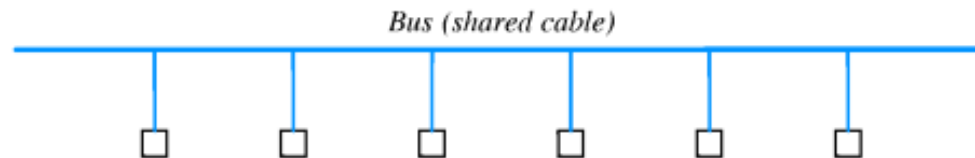
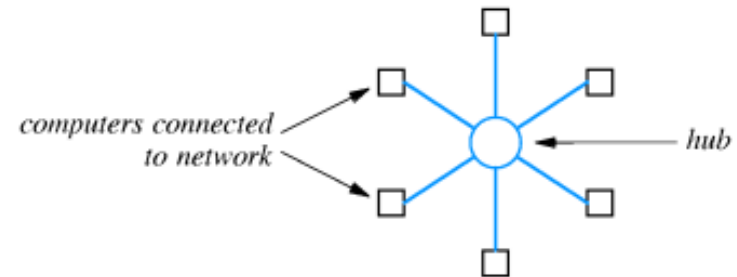
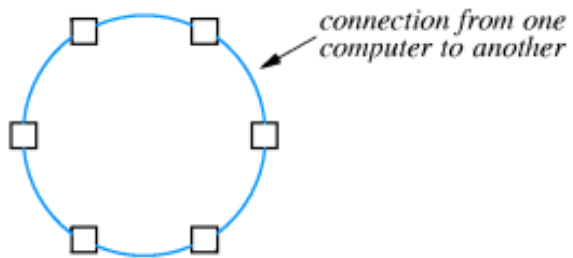
- **Local-Area Network (LAN)** – designed to cover small geographical area
 - Consists of multiple computers (workstations, laptops, mobile devices), peripherals (printers, storage arrays), and routers providing access to other networks
 - Ethernet and/or Wireless (**WiFi**) most common way to construct LANs
 - ▶ Ethernet defined by standard IEEE 802.3 with speeds typically varying from 10Mbps to over 10Gbps
 - ▶ WiFi defined by standard IEEE 802.11 with speeds typically varying from 11Mbps to over 400Mbps.
 - ▶ Both standards constantly evolving

Local-area Network



Local-area Network

- At the beginning, a shared Communication Medium



- Node use a MAC (Medium Access Protocol) to regulate the access to the communication medium

Physical Addressing Scheme

- How to implement unicast communication?



- Each node in the LAN is assigned with a unique address (physical address or hw address or MAC address)
- The sending node inserts the destination address in each packet it sends
- The destination node accepts the received packet *only if* the destination address corresponds to its own address.

An Example of Ethernet LAN

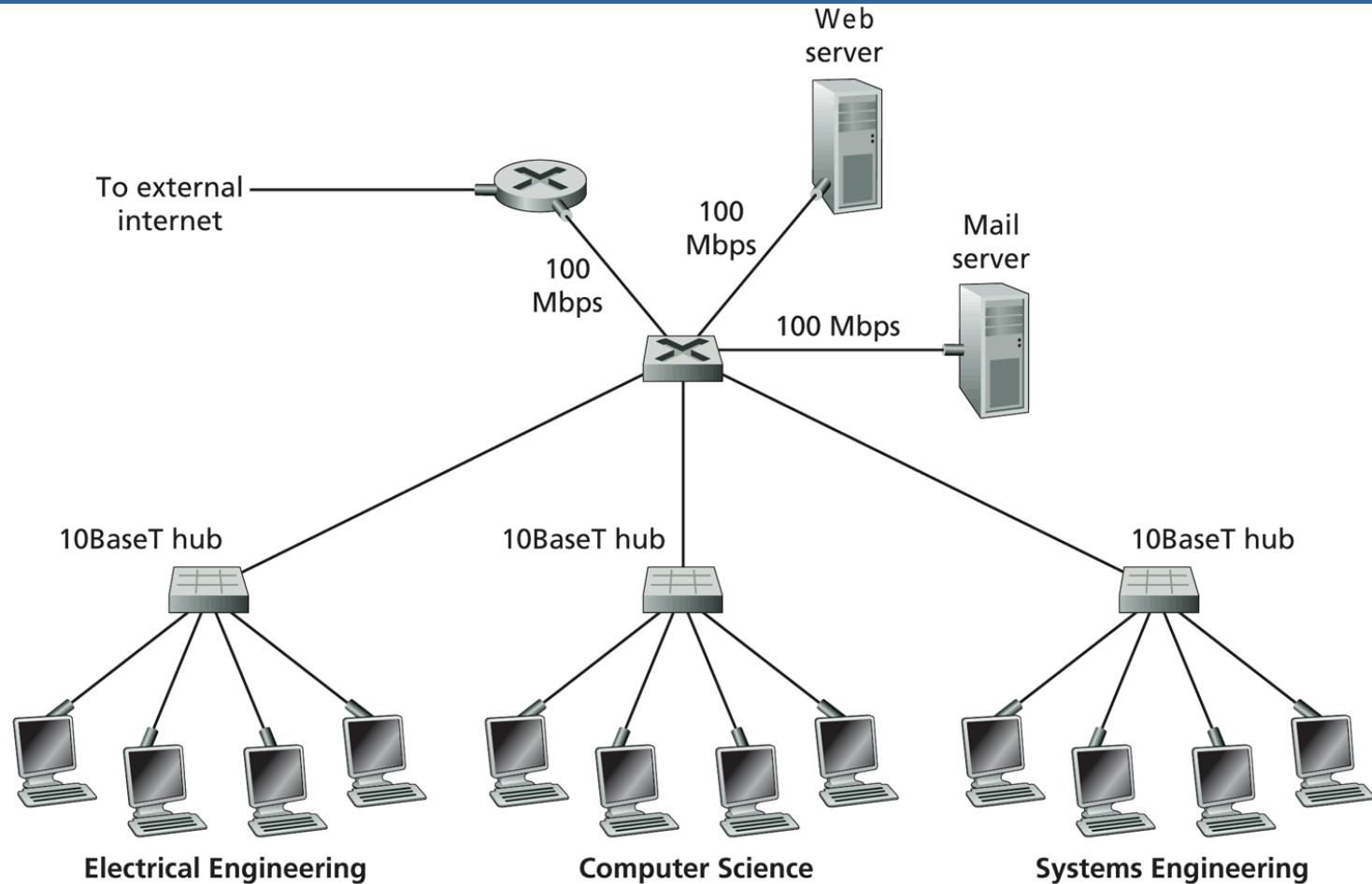
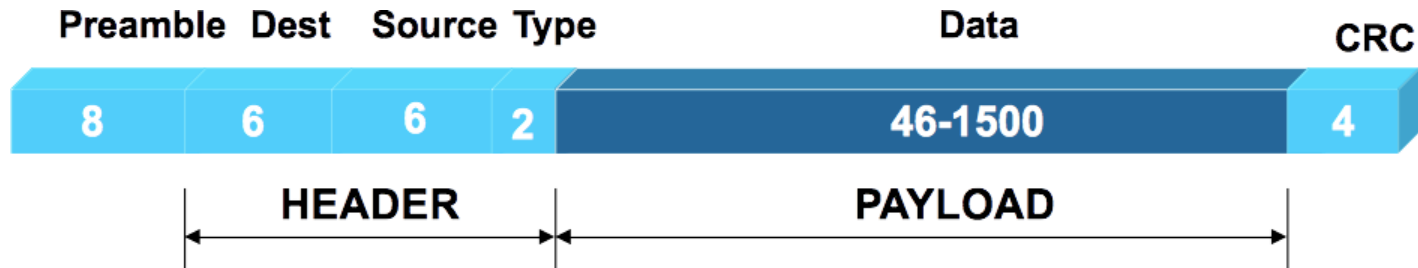


Figure 5.29 ♦ An institutional network using a combination of hubs, Ethernet switches, and a router

Frame Ethernet



- Preamble
 - 64-bit string used for clock synchronization
- Destination Address
 - Broadcast address: 11111.....1
- Source Address
- Type
 - Specify the data type (e.g., 0800: IPv4)
- Payload
- CRC (Cyclic Redundancy Check)
 - Error detection

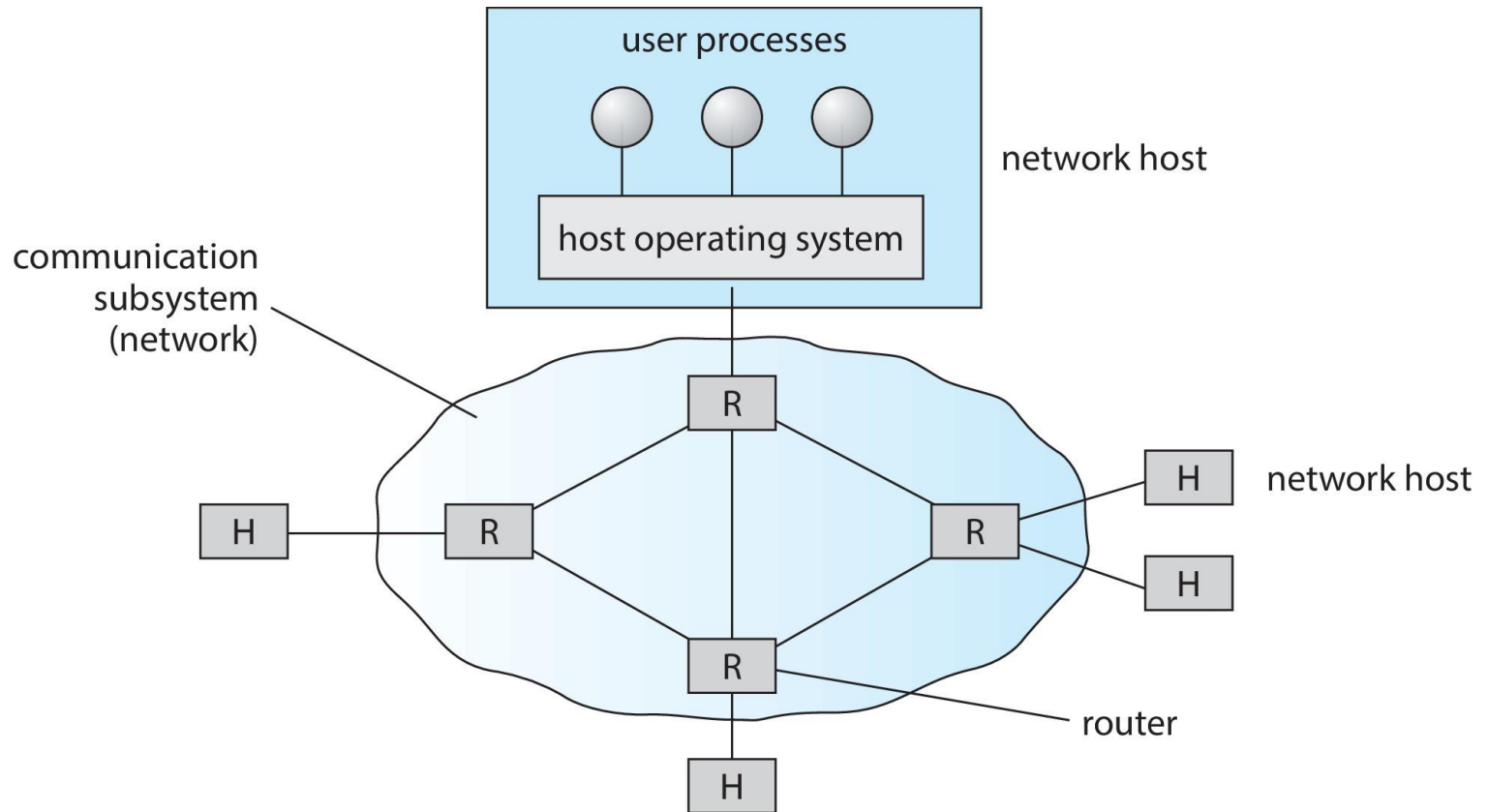
Ethernet MAC Protocol

- **CSMA/CD** - Carrier sense with multiple access (CSMA); collision detection (CD)
 - A node determines whether another packet is currently being transmitted over that link (carrier sense).
 - If the link is sensed as free the packet transmission is started. The node continues listening while transmitting
 - If two or more nodes begin transmitting at exactly the same time, then they will register a collision and will stop transmitting
 - A collided packet is re-tried after a random backoff interval

Network Structure (Cont.)

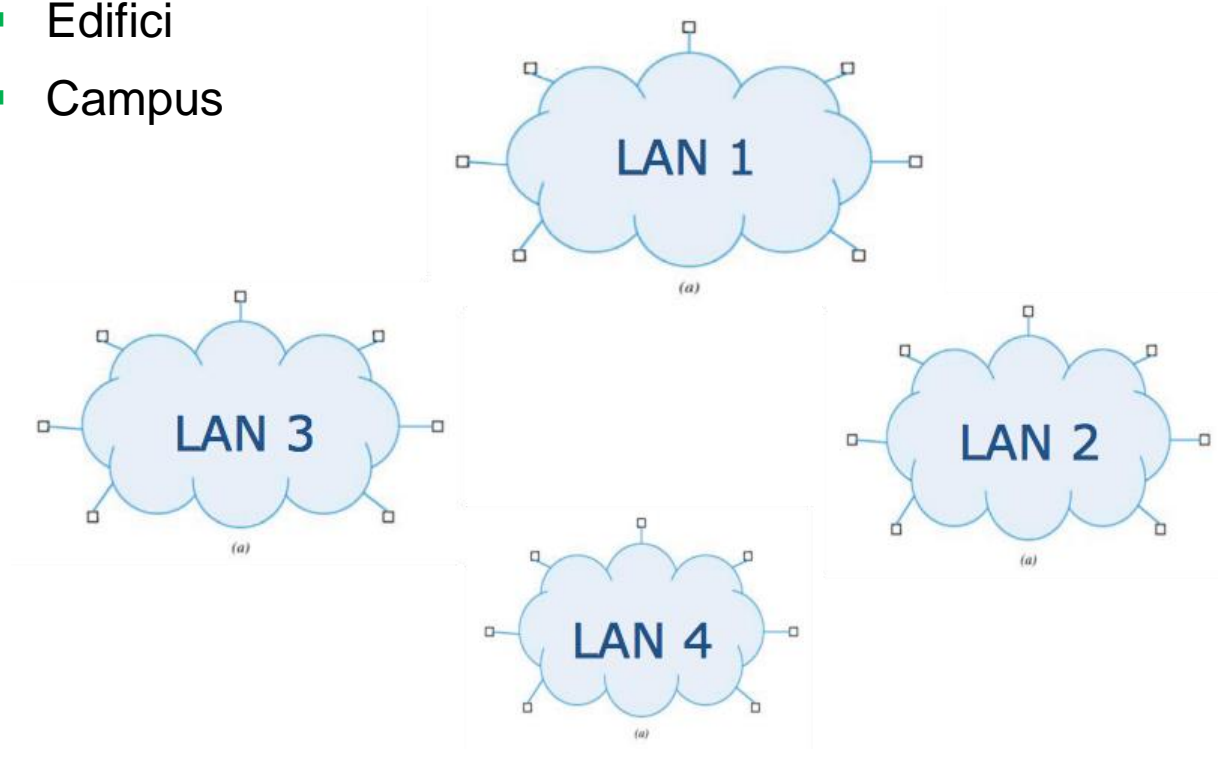
- **Wide-Area Network (WAN)** – links geographically separated sites
 - Point-to-point connections via links
 - ▶ Optical cable, radio waves, and satellite channels
 - Implemented via **routers** to direct traffic from one network to another
 - Internet WAN enables hosts world-wide to communicate
 - Speeds vary
 - ▶ Many backbone providers have speeds at 40-100Gbps
 - ▶ Local **Internet Service Providers (ISPs)** may be slower
 - ▶ WAN links constantly being upgraded
 - WANs and LANs interconnect

Wide-Area Network (WAN)



Limitazioni delle LAN

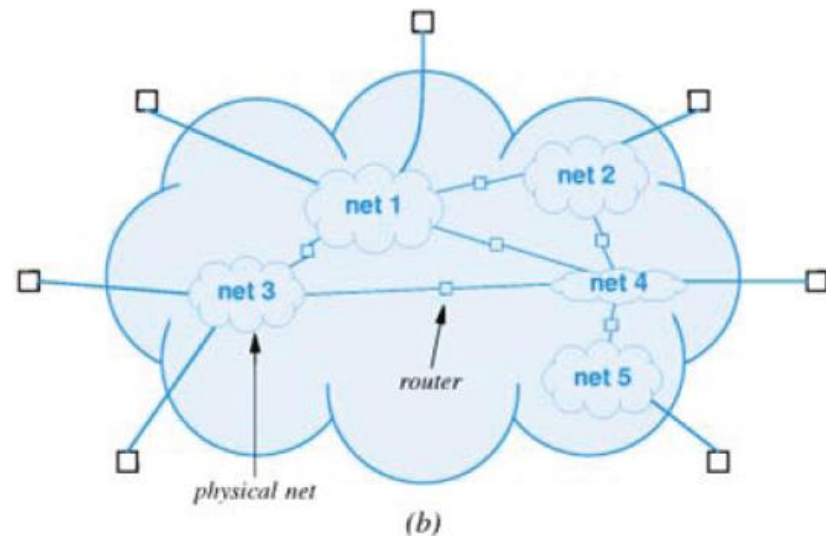
- Le LAN permettono di interconnettere calcolatori in ambito locale
 - Ambienti SOHO
 - Edifici
 - Campus



Come far comunicare fra loro calcolatori collegati a reti di tipo diverso e in posti diversi?

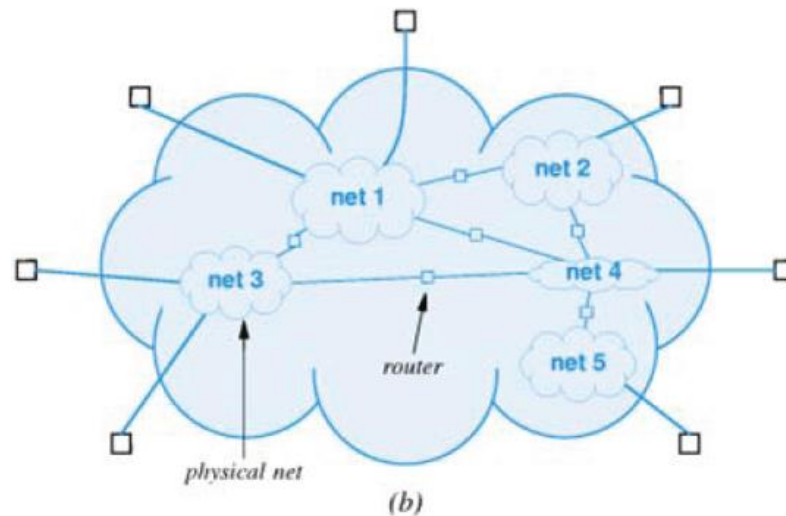
Quello che vorremmo ...

- Due calcolatori devono poter comunicare indipendentemente dalla rete fisica a cui sono collegati
- Aumento di produttività
- Problemi da risolvere
 - Diversi segnali elettrici,
 - Diverso formato dei pacchetti
 - Diverso schema di indirizzamento



Inter-rete

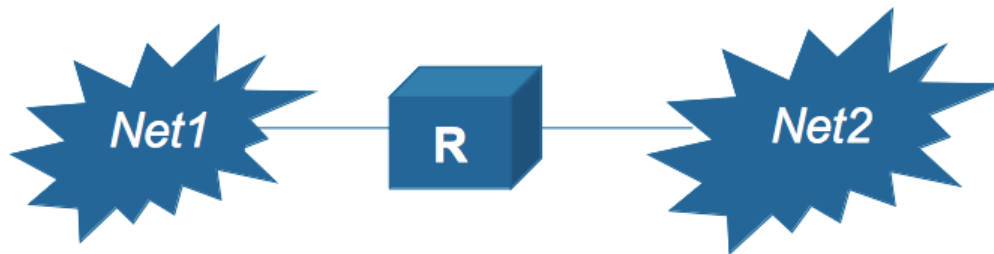
- Occorre
 - Raccordare fisicamente le reti fisiche mediante opportuni dispositivi fisici (router)
 - Aggiungere uno strato software sopra l'hardware di rete
 - ▶ Fa apparire l'insieme di reti eterogenee come un unico sistema



Il sistema che si realizza è detto **inter-rete** o **internet**

Router

- Calcolatore specializzato dedicato alla interconnessione
 - CPU, memoria, ...
 - una interfaccia per ciascuna rete a cui è collegato
- Si collega come un qualsiasi altro calcolatore
 - Collegato contemporaneamente a (almeno) due reti fisiche
- Può interconnettere reti di tipo qualsiasi
- **Inter-rete = reti di calcolatori collegate da router**



Protocolli software

- Simulano una rete virtuale
- Nascondono i dettagli delle reti fisiche sottostanti
 - L'utente si può disinteressare del tipo di reti fisiche sottostanti, della presenza o meno di router, ecc.
- Realizzano un servizio universale
 - Ogni calcolatore è individuato tramite un **indirizzo software**
 - Ogni calcolatore può scambiare messaggi con altri calcolatori collegati alla inter-rete

Protocolli TCP/IP

- TCP = Transmission Control Protocol
- IP = Internet Protocol

- Famiglia di protocolli usati in **Internet**
- Usati anche per la realizzazione di inter-reti private (Intranet)
- Progettati verso gli inizi degli anni '70 su iniziativa del Pentagono
 - Agenzia ARPA → Arpanet
 - Arpanet → Internet

Communication Protocol

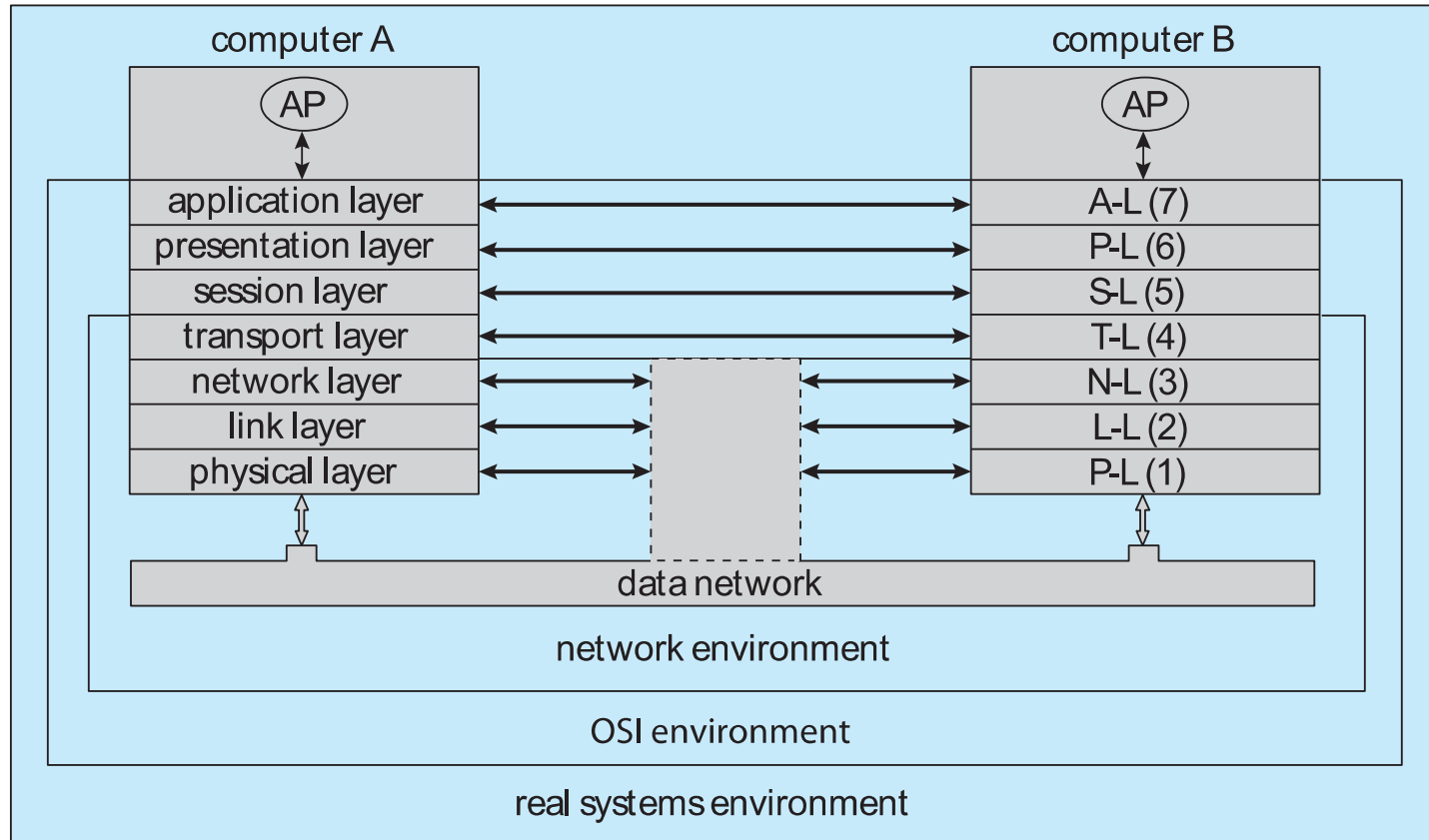
The communication network is partitioned into the following multiple layers:

- **Layer 1: Physical layer** – handles the mechanical and electrical details of the physical transmission of a bit stream
- **Layer 2: Data-link layer** – handles the *frames*, or fixed-length parts of packets, including any error detection and recovery that occurred in the physical layer
- **Layer 3: Network layer** – provides connections and routes packets in the communication network, including handling the address of outgoing packets, decoding the address of incoming packets, and maintaining routing information for proper response to changing load levels

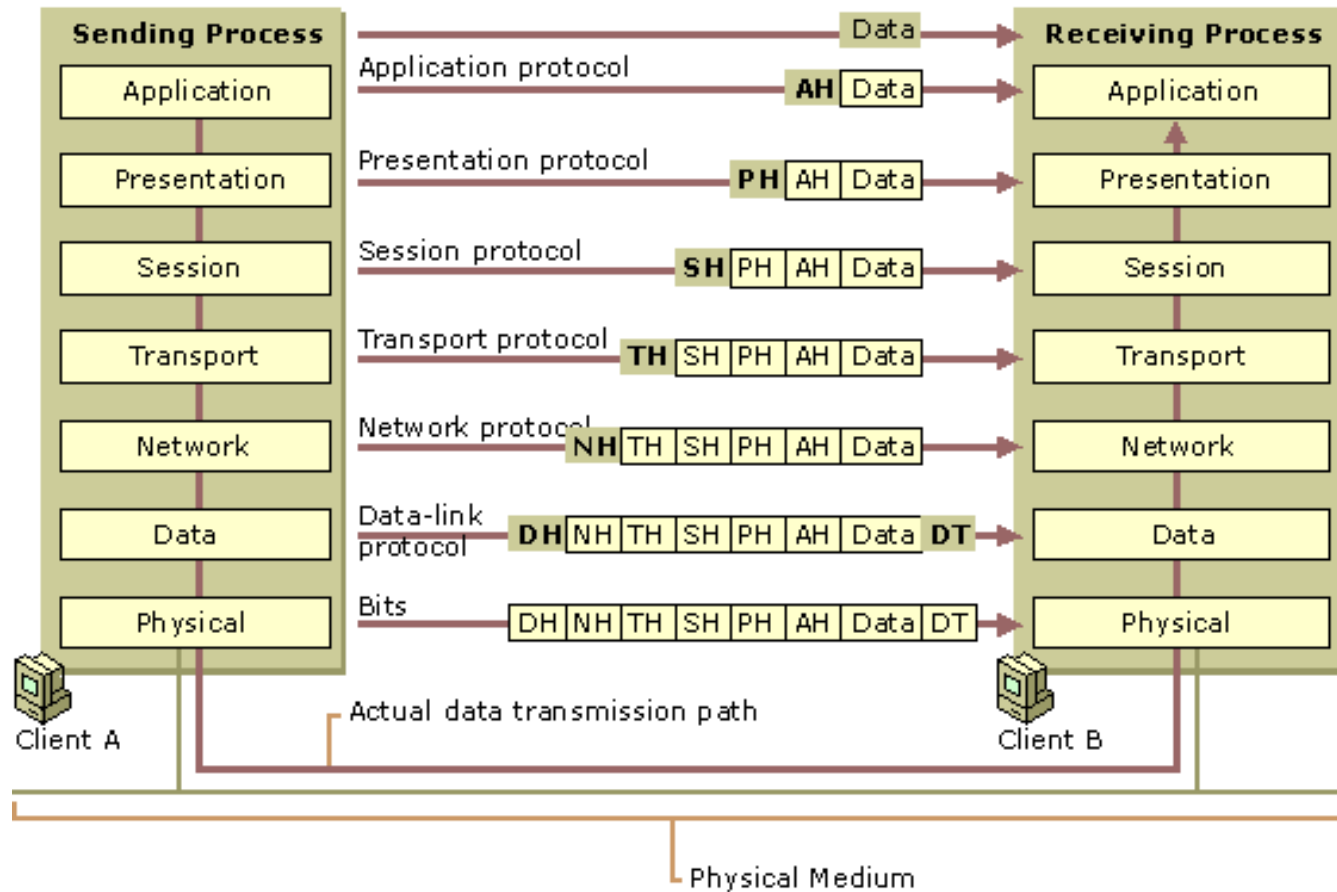
Communication Protocol (Cont.)

- **Layer 4: Transport layer** – responsible for low-level network access and for message transfer between clients, including partitioning messages into packets, maintaining packet order, controlling flow, and generating physical addresses
- **Layer 5: Session layer** – implements sessions, or process-to-process communications protocols
- **Layer 6: Presentation layer** – resolves the differences in formats among the various sites in the network, including character conversions, and half duplex/full duplex (echoing)
- **Layer 7: Application layer** – interacts directly with the users, deals with file transfer, remote-login protocols and electronic mail, as well as schemas for distributed databases

Communication Via ISO Network Model

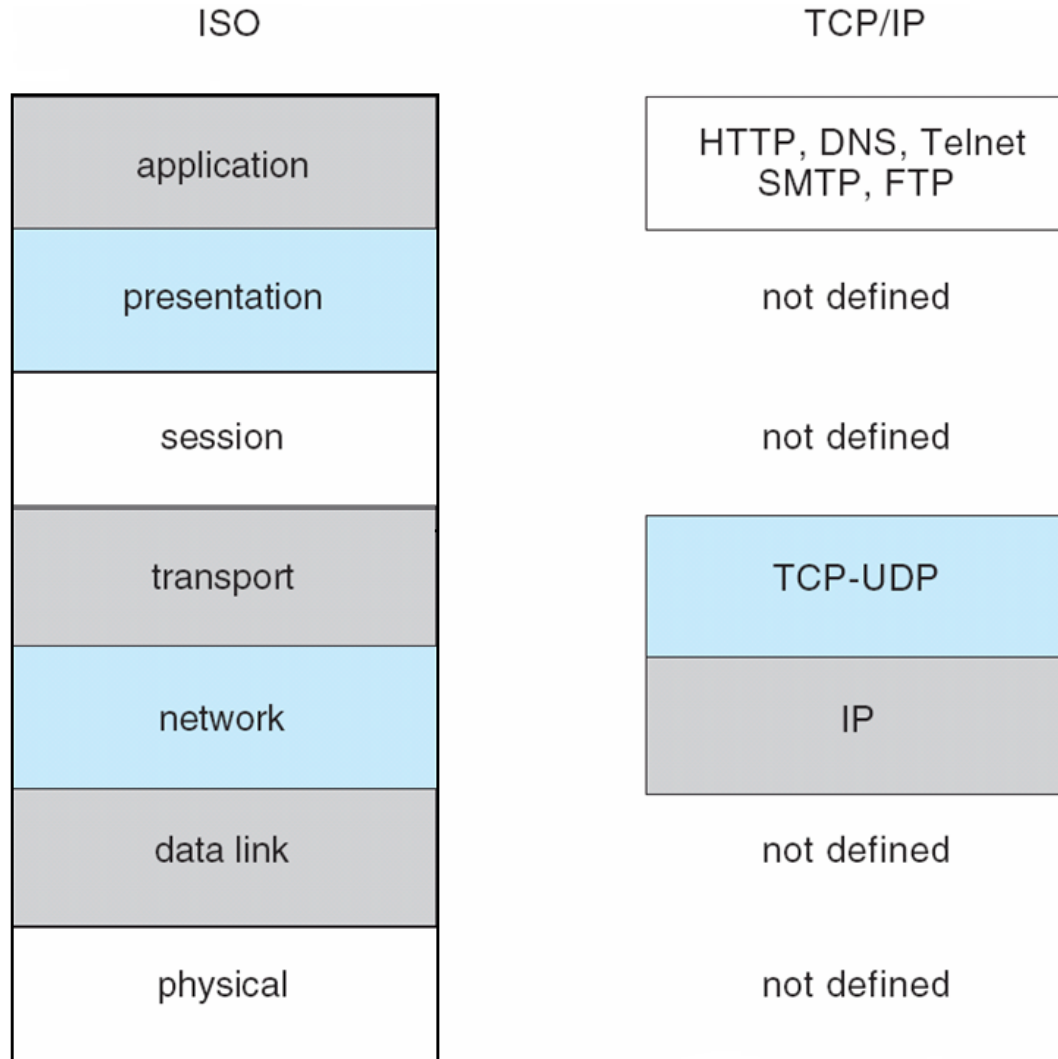


Data path



Picture from <https://technet.microsoft.com/en-us/library/cc977591.aspx>

The TCP/IP Protocol Layers



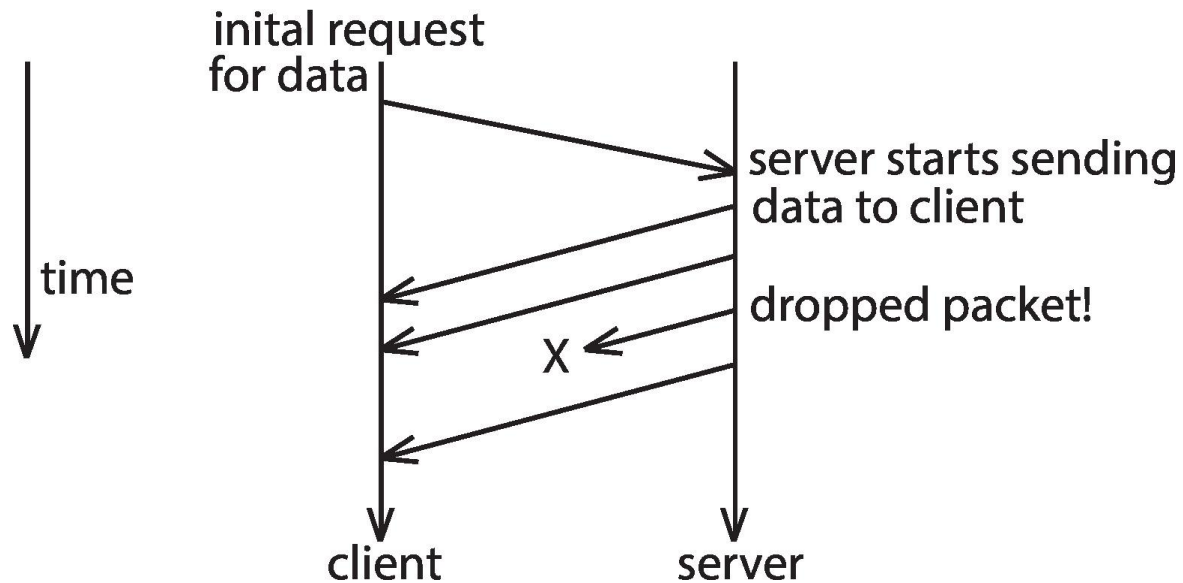
Internet Layer (Protocollo IP)

- Definisce
 - Formato dei pacchetti (datagram)
 - Formato degli indirizzi software (indirizzi IP)
 - Instradamento dei datagrammi
- Servizio best-effort
 - Connectionless
 - ▶ Possibili fuori sequenza
 - Non affidabile
 - ▶ Perdite e/o alterazioni dei datagram
 - ▶ Nessuna garanzia di QoS (ritardo, jitter, throughput)

Transport Layer (Protocollo UDP)

- Demultiplexing dei datagrammi IP
 - Riceve un flusso indistinto di datagrammi IP
 - Recapita i datagrammi ai processi applicativi a cui sono destinati
- Nessun incremento al servizio offerto da IP
 - Non affidabile
 - ▶ La rete sottostante IP non garantisce consegna dei pacchetti -> i pacchetti UDP possono essere persi
 - ▶ Analogamente i pacchetti UDP possono essere ricevuti fuori ordine
 - Connectionless
 - ▶ Non ci sono le fasi di creazione di una connessione e di chiusura
- I pacchetti UDP sono anche detti **datagrammi** (**datagrams**)

UDP Dropped Packet Example

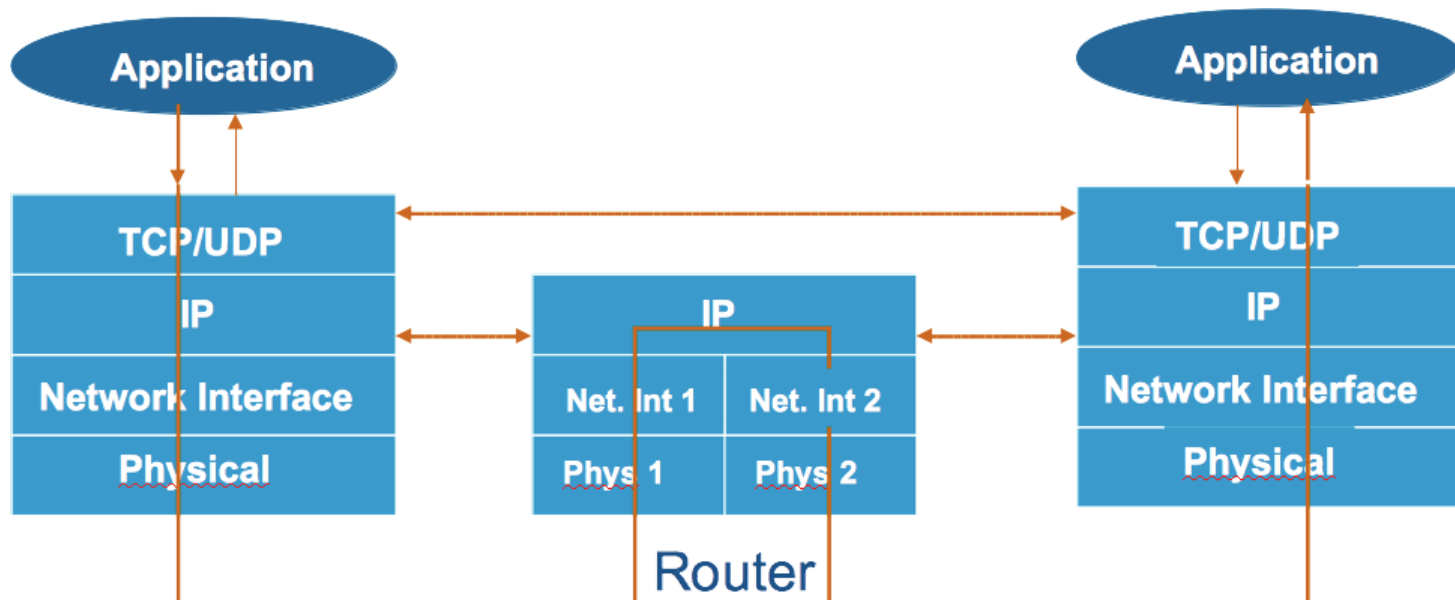


Transport Layer (Protocollo TCP)

- Flusso di byte (stream)
 - ma la comunicazione è sempre a pacchetti (segmenti)
- Trattamento di fuori-sequenza e duplicati
- Rilevazione dei segmenti alterati o persi
- Recupero dei segmenti alterati, persi, ritardati
- Controllo del flusso
- Controllo della congestione
- Servizio **connection-oriented e affidabile**
 - **Tutti** i segmenti vengono consegnati in sequenza
 - Assenza di duplicati
 - Nessuna garanzia sul ritardo, sul jitter e sul throughput

Host, Router e Protocolli

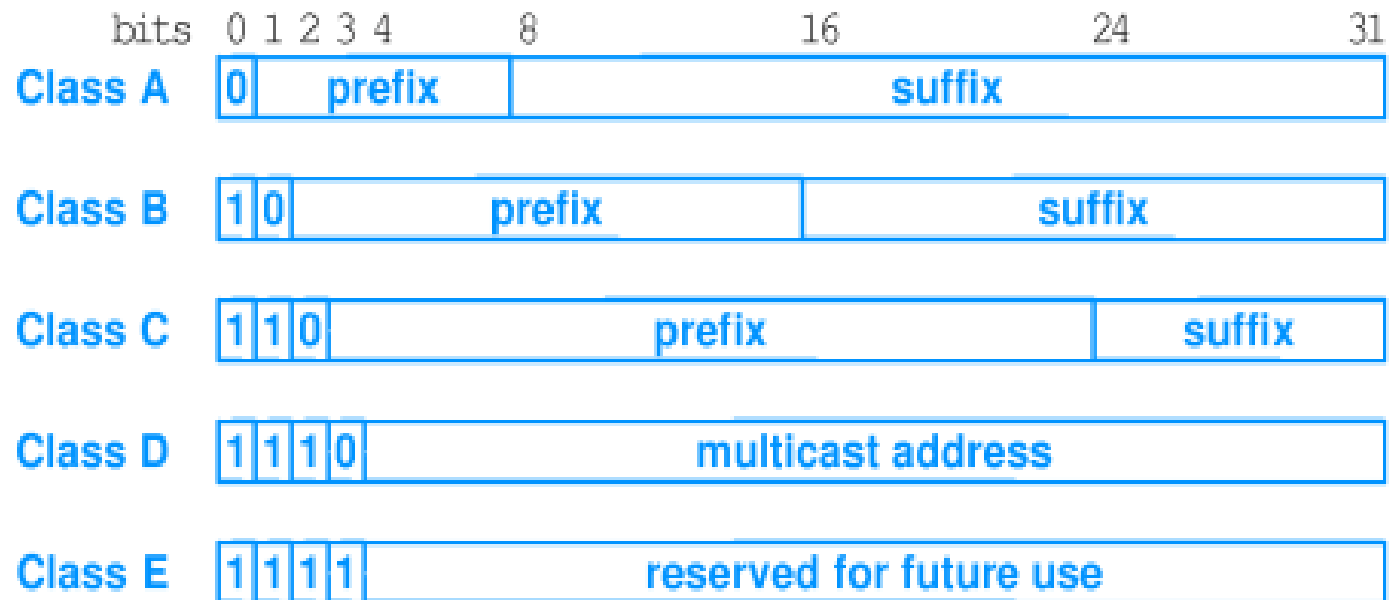
- Host = qualsiasi calcolatore collegato alla inter-rete



Indirizzi IP

- Indirizzo a 32 bit assegnato a ogni host
- Struttura Gerarchica
 - Indirizzo di rete (prefisso) + Indirizzo di host (suffisso)
- Indirizzo di rete (network number)
 - Identifica una rete fisica
 - Assegnato da una autorità centrale che garantisce l'univocità
- Indirizzo di host (host number)
 - Identifica un particolare host all'interno della rete fisica
 - Assegnato localmente dall'amministratore

Classi di indirizzi IP



Notazione decimale puntata

- I 4 byte sono interpretati come numeri decimali
 - compresi fra 0 e 255
- Indirizzo letto come 4 numeri decimali separati da punti

<u>32-bit Binary Number</u>				<u>Equivalent Dotted Decimal</u>
10000001	00110100	00000110	00000000	129 . 52 . 6 . 0
11000000	00000101	00110000	00000011	192 . 5 . 48 . 3
00001010	00000010	00000000	00100101	10 . 2 . 0 . 37
10000000	00001010	00000010	00000011	128 . 10 . 2 . 3
10000000	10000000	11111111	00000000	128 . 128 . 255 . 0

Indirizzi IP riservati

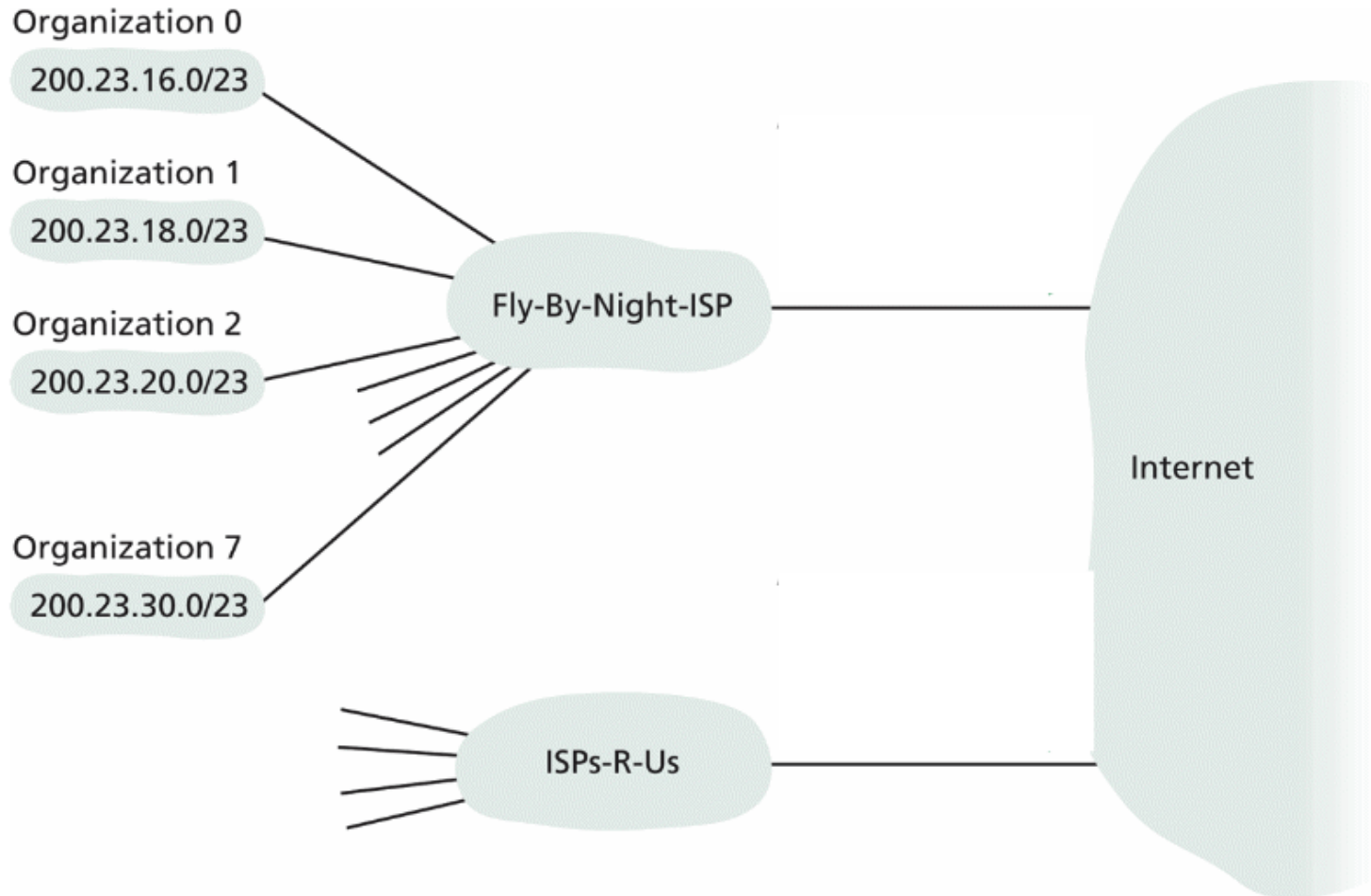
- Indirizzo per l'intera sottorete
 - Suffisso di tutti zeri (131.114.0.0)
- Indirizzo di trasmissione broadcast orientata
 - Suffisso di tutti 1 (131.114.255.255)
- Indirizzo di trasmissione broadcast ristretta
 - Costituito da tutti 1 (255.255.255.255)
- Indirizzo di “questo calcolatore”
 - Indirizzi di tutti 0
 - Usato all'avvio
- Indirizzo loopback
 - 127.0.0.1
 - Usato nello fase di sviluppo di applicazione di rete

IP addressing: CIDR

- **CIDR: Classless InterDomain Routing**
 - subnet portion of address of arbitrary length
 - address format: **a.b.c.d/x** , where x is # bits in subnet portion of address



Hierarchical Addressing



Hierarchical addressing

IP Addresses: how to get one?

- Q: How does *network* get subnet part of IP addr?
- A: gets allocated portion of its provider ISP's address space

ISP's block	<u>11001000</u>	<u>00010111</u>	<u>00010000</u>	00000000	200.23.16.0/20
Organization 0	<u>11001000</u>	<u>00010111</u>	<u>00010000</u>	00000000	200.23.16.0/23
Organization 1	<u>11001000</u>	<u>00010111</u>	<u>00010010</u>	00000000	200.23.18.0/23
Organization 2	<u>11001000</u>	<u>00010111</u>	<u>00010100</u>	00000000	200.23.20.0/23
...	
Organization 7	<u>11001000</u>	<u>00010111</u>	<u>00011110</u>	00000000	200.23.30.0/23

IP addressing: the last word...

- Q: How does an ISP get block of addresses?
- A: **ICANN**: Internet Corporation for Assigned Names and Numbers
 - allocates addresses
 - manages DNS
 - assigns domain names, resolves disputes

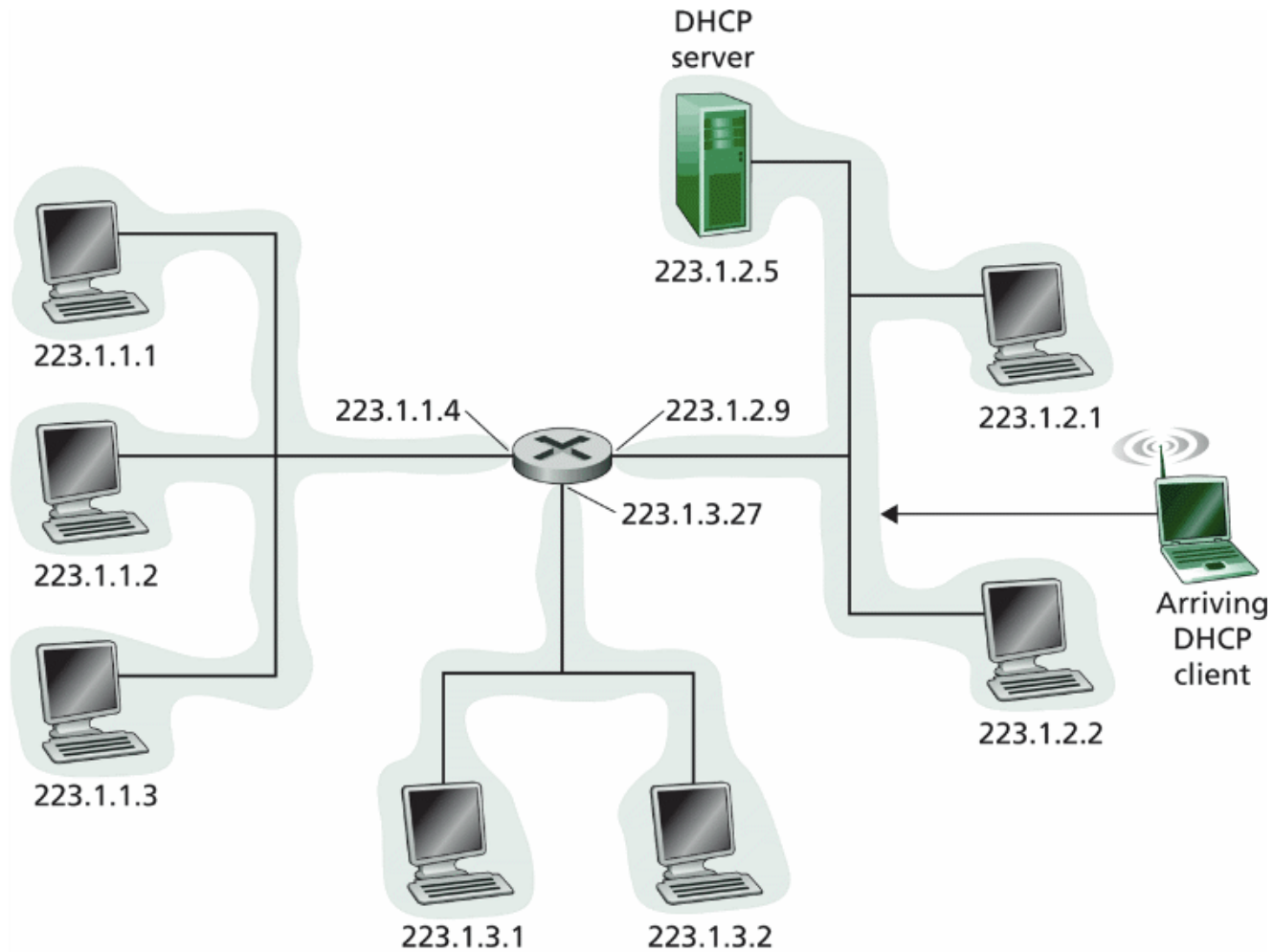
IP addresses: how to get one?

- Q: How does a *host* get an IP address?
- **Permanent Address**
 - hard-coded by system admin in a file
 - Windows: control-panel->network->configuration->tcp/ip->properties
 - UNIX: /etc/rc.config
- **Temporary Address**
 - **DHCP**: **D**ynamic **H**ost **C**onfiguration **P**rotocol: dynamically get address from as server
 - “plug-and-play”

DHCP: Dynamic Host Configuration Protocol

- Goal: allow host to *dynamically* obtain its IP address from network server when it joins network
 - Allows reuse of addresses (only hold address while connected “on”)
 - Support for mobile users who want to join network (more shortly)
- DHCP overview:
 - host broadcasts “DHCP discover ” msg
 - DHCP server responds with “DHCP offer ” msg
 - host requests IP address: “DHCP request ” msg
 - DHCP server sends address: “DHCP ack ” msg

DHCP client-server scenario

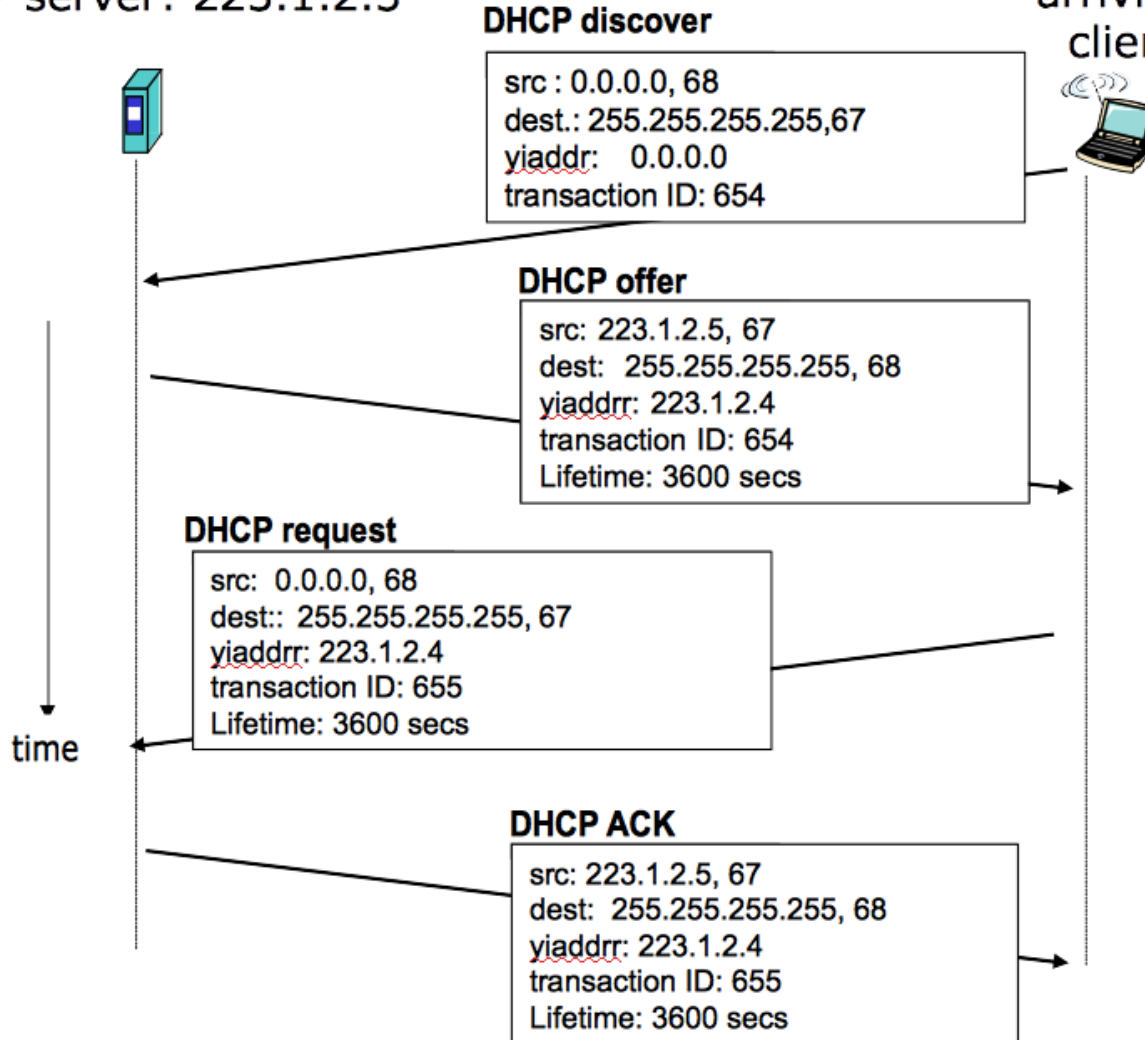


DHCP client-server scenario

DHCP client-server scenario

DHCP server: 223.1.2.5

arriving
client



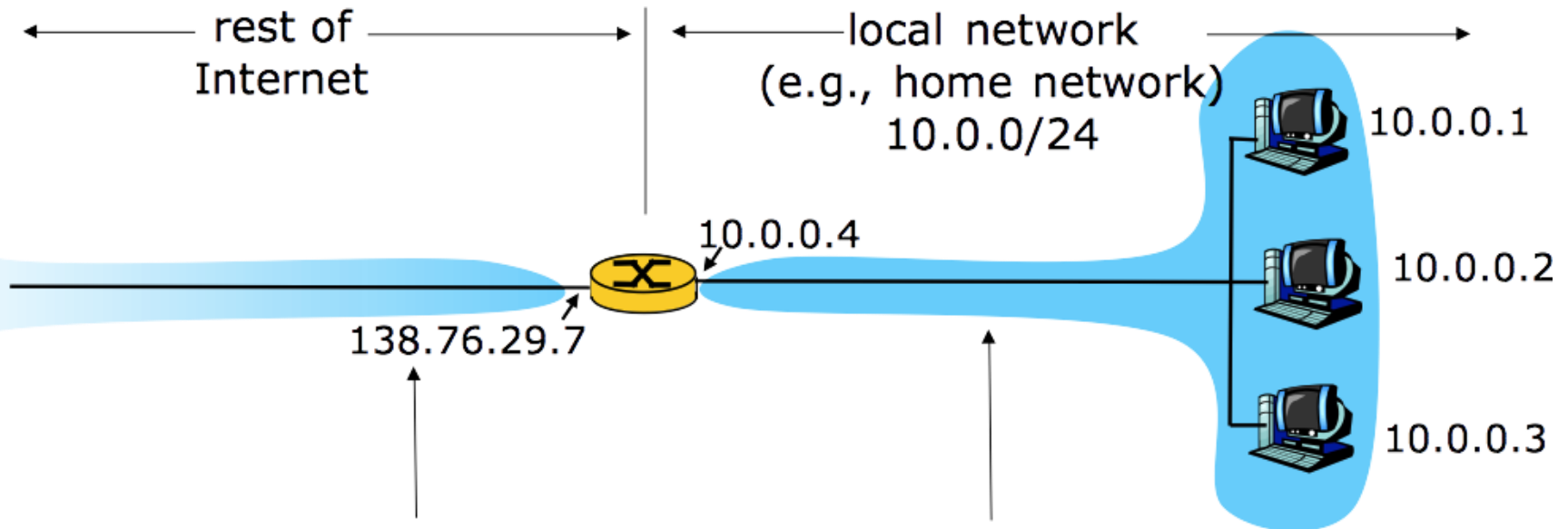
DHCP: more than IP address

- DHCP can return more than just allocated IP address on subnet:
 - address of first-hop router for client
 - name and IP address of DNS sever
 - network mask (indicating network versus host portion of address)

NAT: Network Address Translation

- **Motivation:** local network uses just one IP address as far as outside world is concerned:
 - range of addresses not needed from ISP: just one IP address for all devices
 - can change addresses of devices in local network without notifying outside world
 - can change ISP without changing addresses of devices in local network
 - devices inside local net not explicitly addressable, visible by outside world (a security plus).

NAT: Network Address Translation



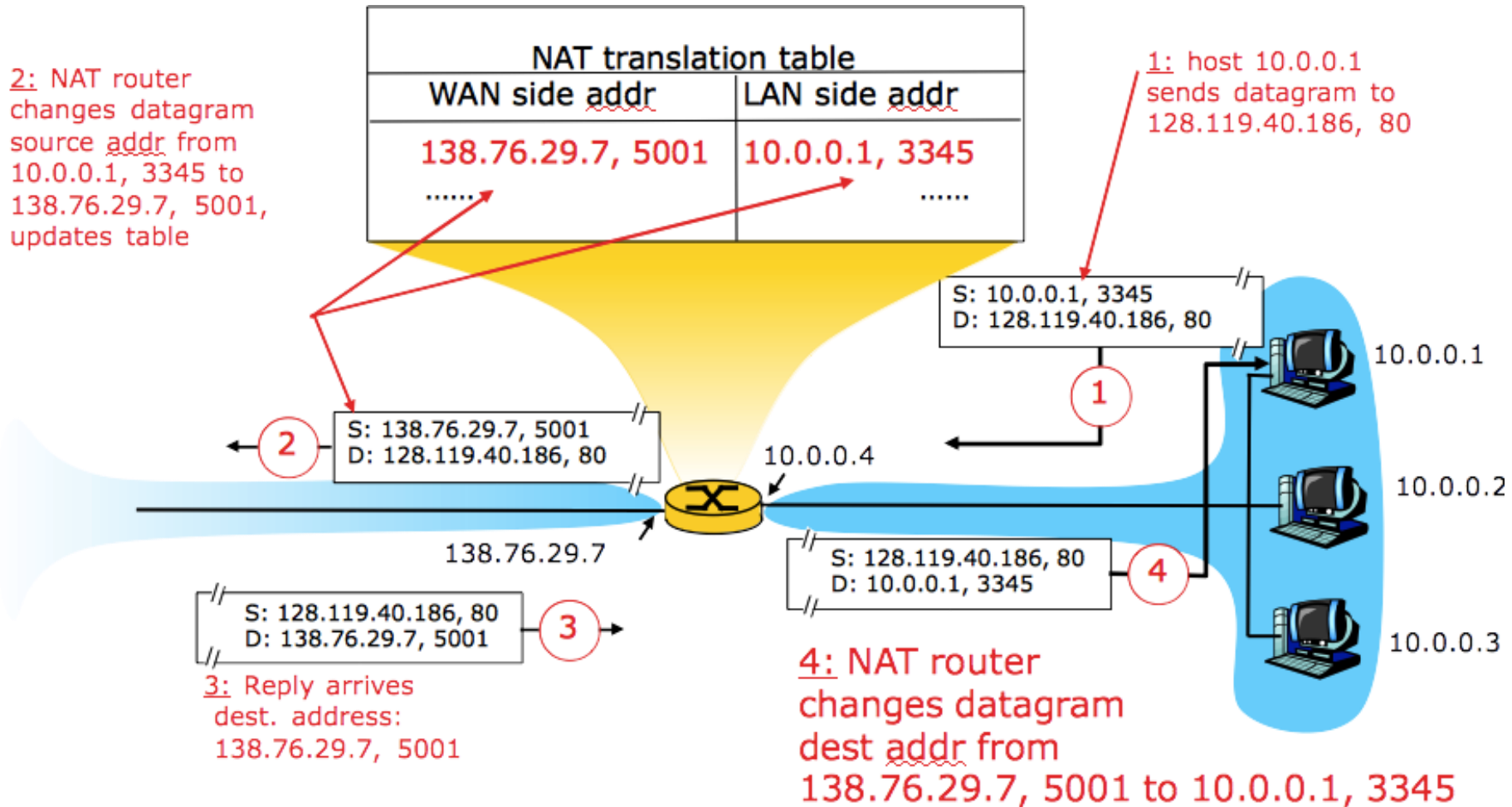
All datagrams *leaving* local network have **same** single source NAT IP address: 138.76.29.7, different source port numbers

Datagrams with source or destination in this network have 10.0.0/24 address for source, destination (as usual)

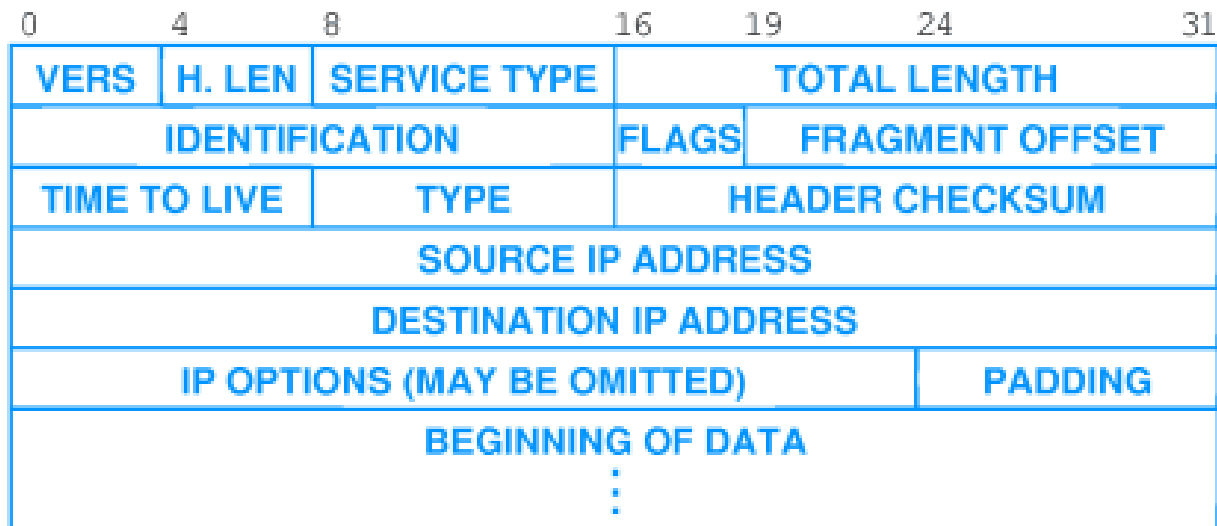
NAT: Network Address Translation

- **Implementation:** NAT router must:
 - *outgoing datagrams: replace* (source IP address, port #) of every outgoing datagram to (NAT IP address, new port #)
... remote clients/servers will respond using (NAT IP address, new port #) as destination addr.
 - *remember (in NAT translation table)* every (source IP address, port #) to (NAT IP address, new port #) translation pair
 - *incoming datagrams: replace* (NAT IP address, new port #) in dest fields of every incoming datagram with corresponding (source IP address, port #) stored in NAT table

NAT: Network Address Translation



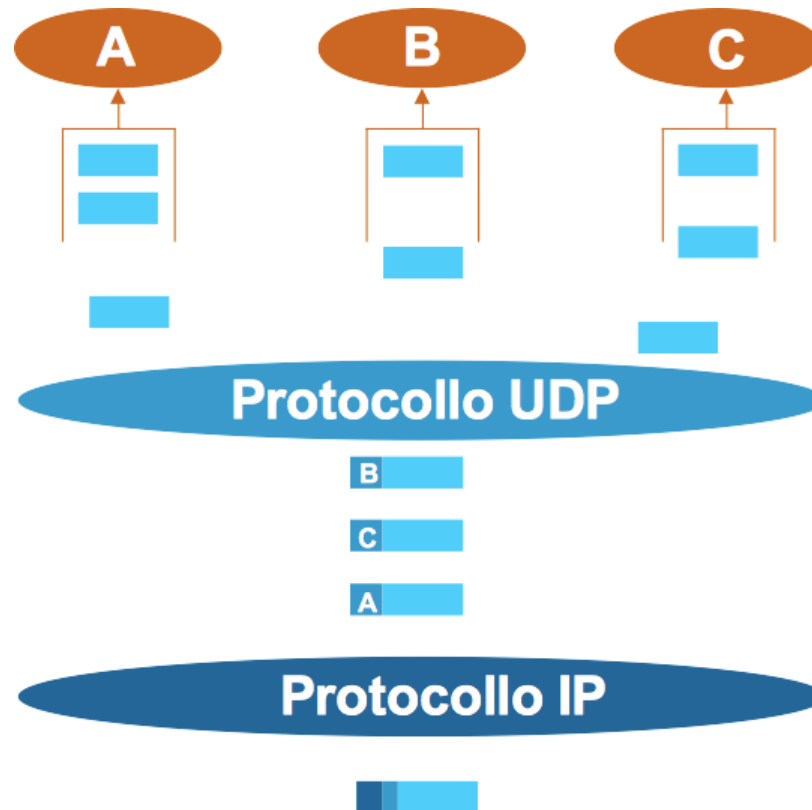
Datagram IP



Livello di trasporto

- Estende il servizio di trasporto host-to-host in un servizio di comunicazione fra processi
 - Esegue il demultiplexing delle informazioni
 - Basato sul concetto di porta
- I processi vengono individuati mediante la coppia
<Host IP Address, Port Number>
- Il processo mittente deve specificare sia l'indirizzo IP che il numero di porta
- Il SO realizza la porta come coda di messaggi

Protocollo UDP



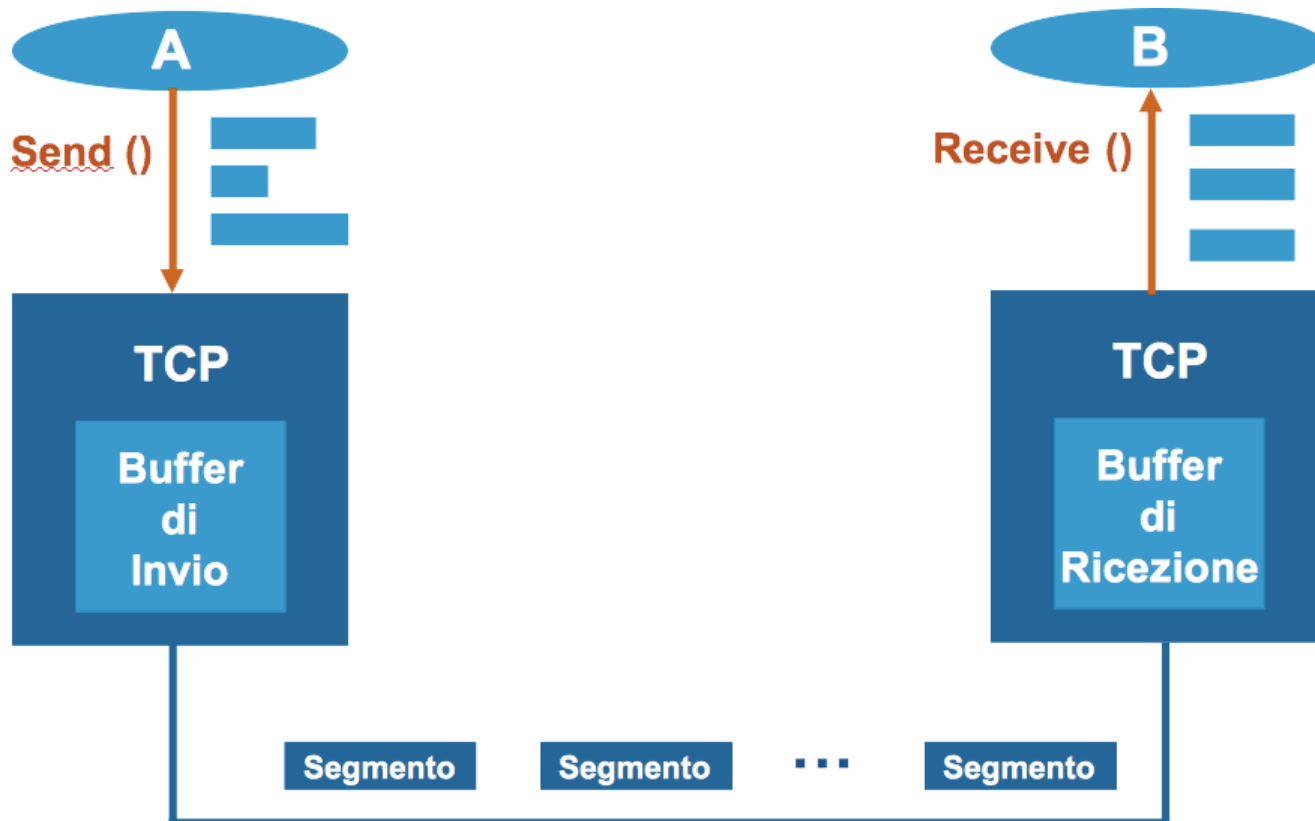
Introduzione al TCP

- Servizio di comunicazione fra processi orientato alla connessione
 - Meccanismo di de-multiplexing basato sulle porte
 - Apertura connessione, trasferimento dati, chiusura connessione
- Punto-Punto
 - Ogni connessione TCP collega esattamente due processi
- Affidabile
 - Consegna affidabile, senza duplicati e in sequenza di un flusso di byte
- Full duplex
 - Un flusso di byte in ciascuna direzione

Introduzione al TCP

- Controllo e gestione degli errori
 - Checksum e ACK
 - Timeout e ritrasmissione
- Controllo del flusso
 - Evita che il mittente invii più dati di quanti il ricevitore possa gestire
- Controllo della congestione
 - Evita che il mittente invii più dati di quanti la rete sia in grado di trasportare

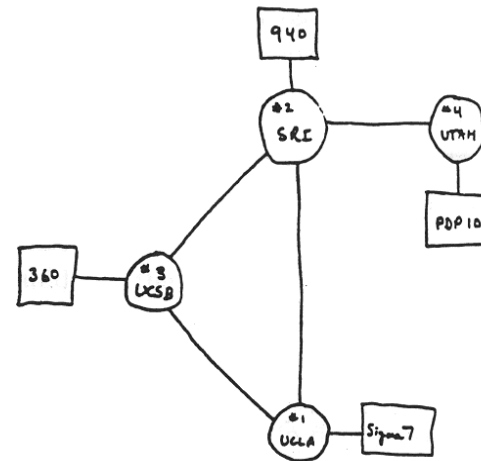
Flusso di byte



Internet History

1961-1972: Early packet-switching principles

- 1961: Kleinrock - queueing theory shows effectiveness of packet-switching
- 1964: Baran - packet-switching in military nets
- 1967: ARPAnet conceived by Advanced Research Projects Agency
- 1969: first ARPAnet node operational
- 1972:
 - ARPAnet public demonstration
 - NCP (Network Control Protocol) first host-host protocol
 - first e-mail program
 - ARPAnet has 15 nodes



THE ARPA NETWORK

Internet History

1972-1980: Internetworking, new and proprietary nets

- 1970: ALOHAnet satellite network in Hawaii
- 1974: Cerf and Kahn - architecture for interconnecting networks
- 1976: Ethernet at Xerox PARC
- late70's: proprietary architectures: DECnet, SNA, XNA
- late 70's: switching fixed length packets (ATM precursor)
- 1979: ARPAnet has 200 nodes

Cerf and Kahn's internetworking principles:

- minimalism, autonomy - no internal changes required to interconnect networks
- best effort service model
- stateless routers
- decentralized control

define today's Internet architecture

Internet History

1980-1990: new protocols, a proliferation of networks

- 1983: deployment of TCP/IP
- 1982: smtp e-mail protocol defined
- 1983: DNS defined for name-to-IP-address translation
- 1985: ftp protocol defined
- 1986: first Italian node connected to the Internet (Pisa)
- 1988: TCP congestion control
- new national networks: Csnet, BITnet, NSFnet, Minitel
- 100,000 hosts connected to confederation of networks

Internet History

- First Italian node connected to the Internet
 - 30 Aprile 1986
 - Nodo CNUCE, Pisa

Consiglio Nazionale delle Ricerche
Istituto CNUCE

Pisa 12/5/86

Prot. n. 1922

Prof. L. Rossi Bernardi
Presidente del CNR

Prof. G. Biorci
Presidente CGI

Ing. S. Trumpy
Direttore CNUCE

Oggetto: Collegamento del CNUCE ad ARPANET.

Dal 30 Aprile scorso, il sistema di calcolo dell'Istituto CNUCE è stato collegato alla rete di elaboratori USA denominata ARPANET (Advanced Research Projects Agency NETWORK). Tale rete, sponsorizzata dal Dipartimento della Difesa USA, collega ormai parecchie migliaia di elaboratori eterogenei per dimensione e per costruttore, operanti presso i più prestigiosi Centri di Ricerca, Università ed Istituzioni Militari prevalentemente USA.

Dopo la Norvegia, la Gran Bretagna e la Germania Ovest, l'Italia è la quarta nazione europea dotata di accesso ad ARPANET tramite la sottorete via satellite SATNET (SATellite NETWORK) (Allegato 1).

A livello italiano tale collegamento è il risultato di una cooperazione tra CNR, TELESPAZIO ed ITALCABLE; cooperazione sancita dalla stipula di un comune contratto triennale che scadrà nell'Agosto del 1987.

Voluta dalla CGI per attuare la politica del calcolo scientifico dell'Ente, ARPANET consente adesso all'utente scientifico italiano collegato al CNUCE di accedere ai servizi disponibili presso gli altri elaboratori della rete e viceversa.

Data la riservatezza dei dati residenti presso gli elaboratori che operano soprattutto in ambito militare, e in conformità alle norme del suddetto contratto, l'accesso dell'utente CNUCE dovrà essere autorizzato, oltre che dal CNR, anche da TELESPAZIO, ITALCABLE e della DARPA (Defence Advanced Research Projects Agency).

Cordiali Saluti

Luciano Lenzini



PS: Allego copia della bozza del comunicato stampa il cui contenuto deve essere concordato con TELESPAZIO ed ITALCABLE.

CNUCE

36 VIA S. MARIA - 56100 - PISA
Tel. (050) 593111 Telex 500371-CNUCE
Telegrammi CNUCE-PISA

Internet History

1990, 2000's: commercialization, the Web, new apps

- **Early 1990's:** ARPAnet decommissioned
- **1991:** NSF lifts restrictions on commercial use of NSFnet (decommissioned, 1995)
- **early 1990s:** Web
 - hypertext [Bush 1945, Nelson 1960's]
 - HTML, HTTP: Berners-Lee
 - 1994: Mosaic, later Netscape
 - late 1990's: commercialization of the Web
- **Late 1990's – 2000's:**
 - more killer apps: instant messaging, P2P file sharing
 - network security to forefront
 - est. 50 million host, 100 million+ users
 - backbone links running at Gbps

Distributed programming

- Distributed Programming
 - Socket interface
 - Client-server Model
 - Peer-to-Peer Model

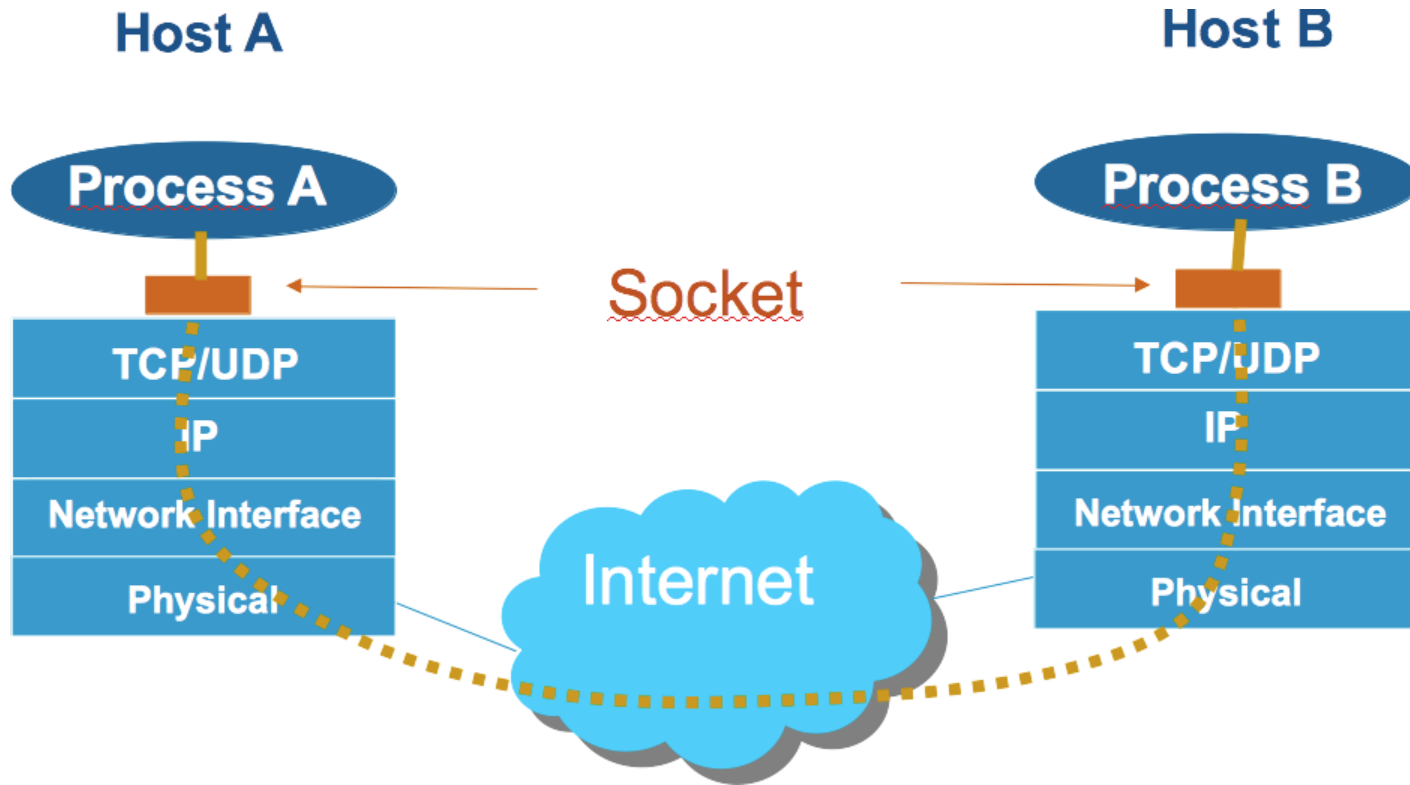
Interfaccia Socket

- Standard *de facto* per la comunicazione fra processi in ambiente distribuito
 - Si può usare anche per la comunicazione fra processi sulla stessa macchina
- Interfaccia unica per operare con i vari protocolli di rete a disposizione
- Nasconde tutti i meccanismi di comunicazione di livello inferiore (TCP/UDP, IP, ...)

Socket

- Estremità di canale di comunicazione identificata da un indirizzo
 - Socket: presa telefonica
 - Indirizzo: numero di telefono
- Indirizzo
 - Indirizzo dell'Host (**Indirizzo IP**)
 - Indirizzo del processo (**Port Number**)
- La comunicazione avviene tramite una coppia di socket

Comunicazione mediante socket



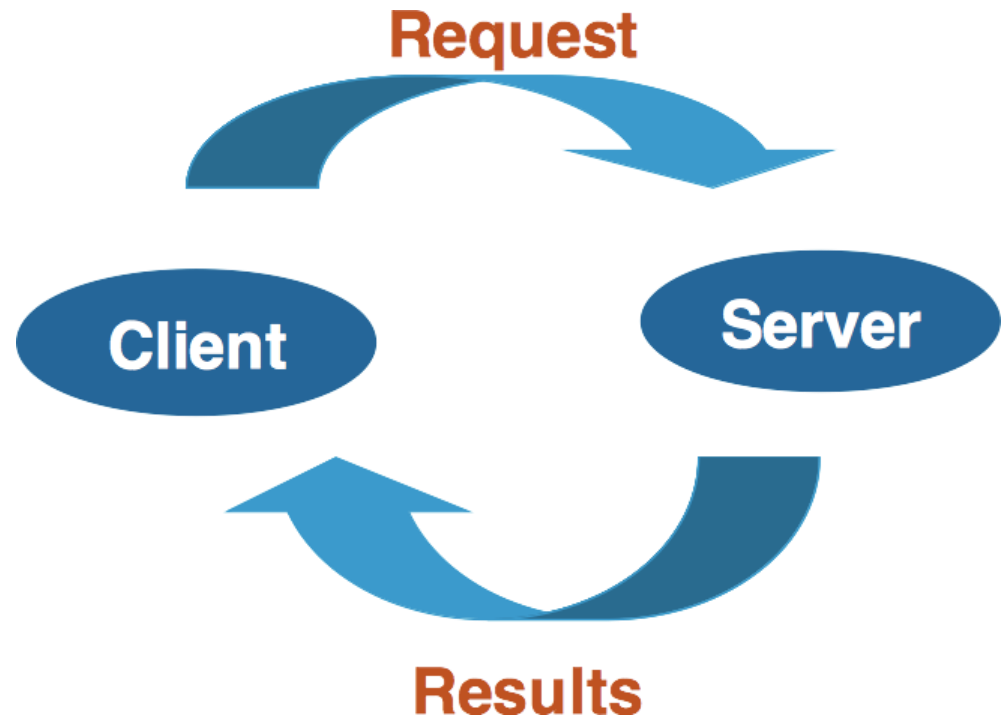
Supporto del SO

- Il SO implementa l'astrazione socket
- System call per
 - Creare un socket
 - Associare indirizzo IP e porta al socket
 - Mettere in ascolto un processo su un socket (server)
 - Accettare una richiesta di servizio su un socket (server)
 - Aprire una connessione verso un socket remoto (client)
 - Inviare un messaggio verso un socket remoto
 - Ricevere un messaggio da un socket
 -

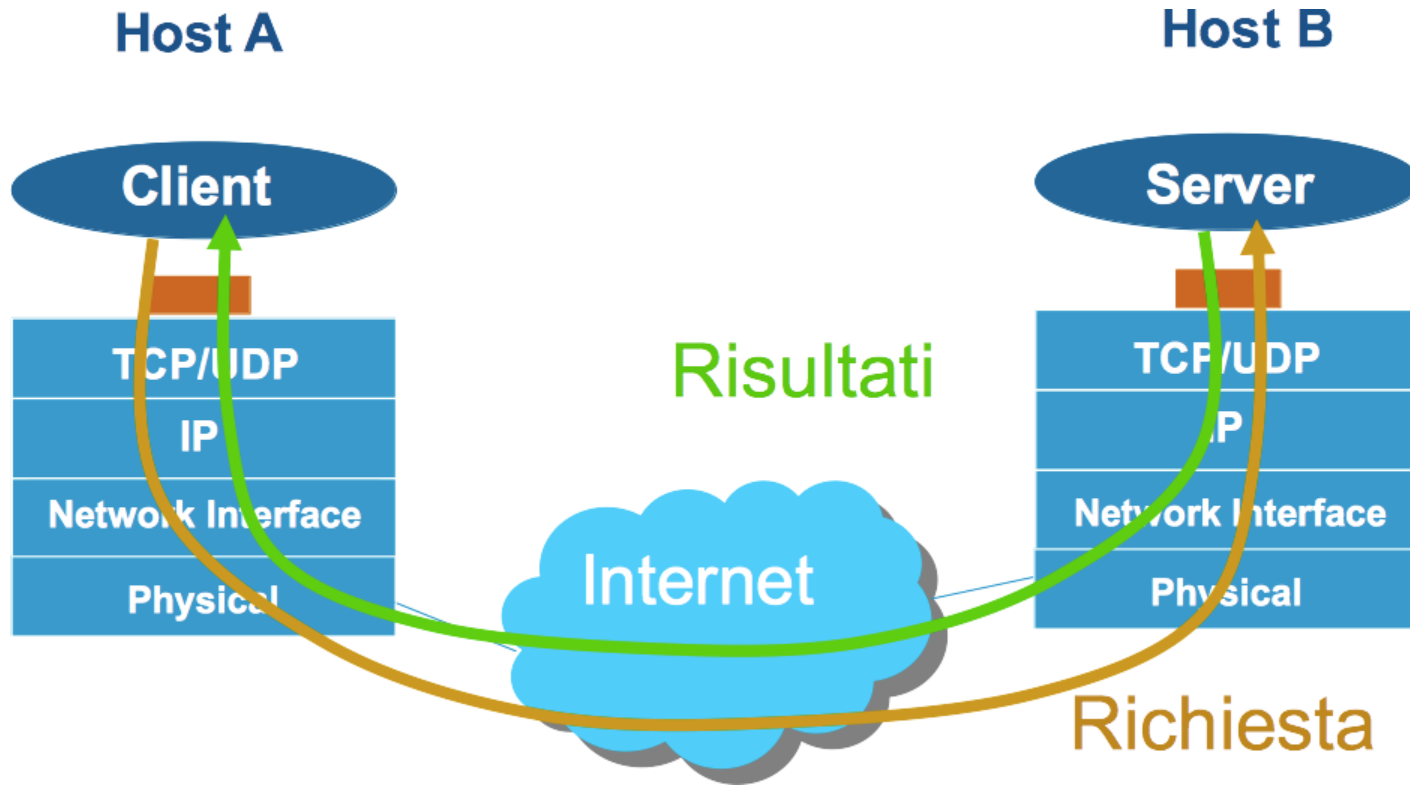
Modello Client-Server

- Paradigma basato su scambio di messaggi
 - Paradigma generale
 - Ma usato principalmente in ambito distribuito

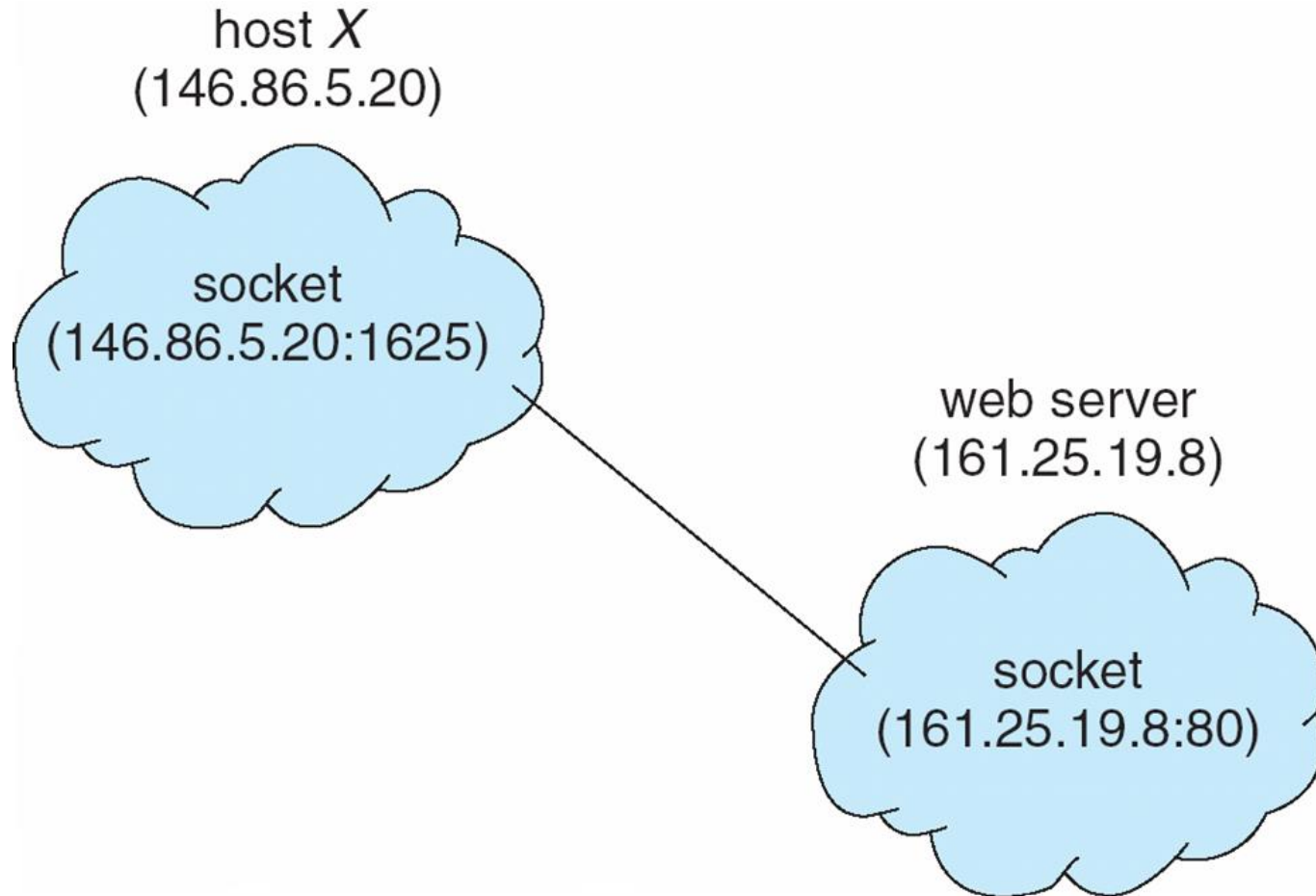
- Scambio di msg per
 - Richiesta di servizio
 - Invio dei risultati



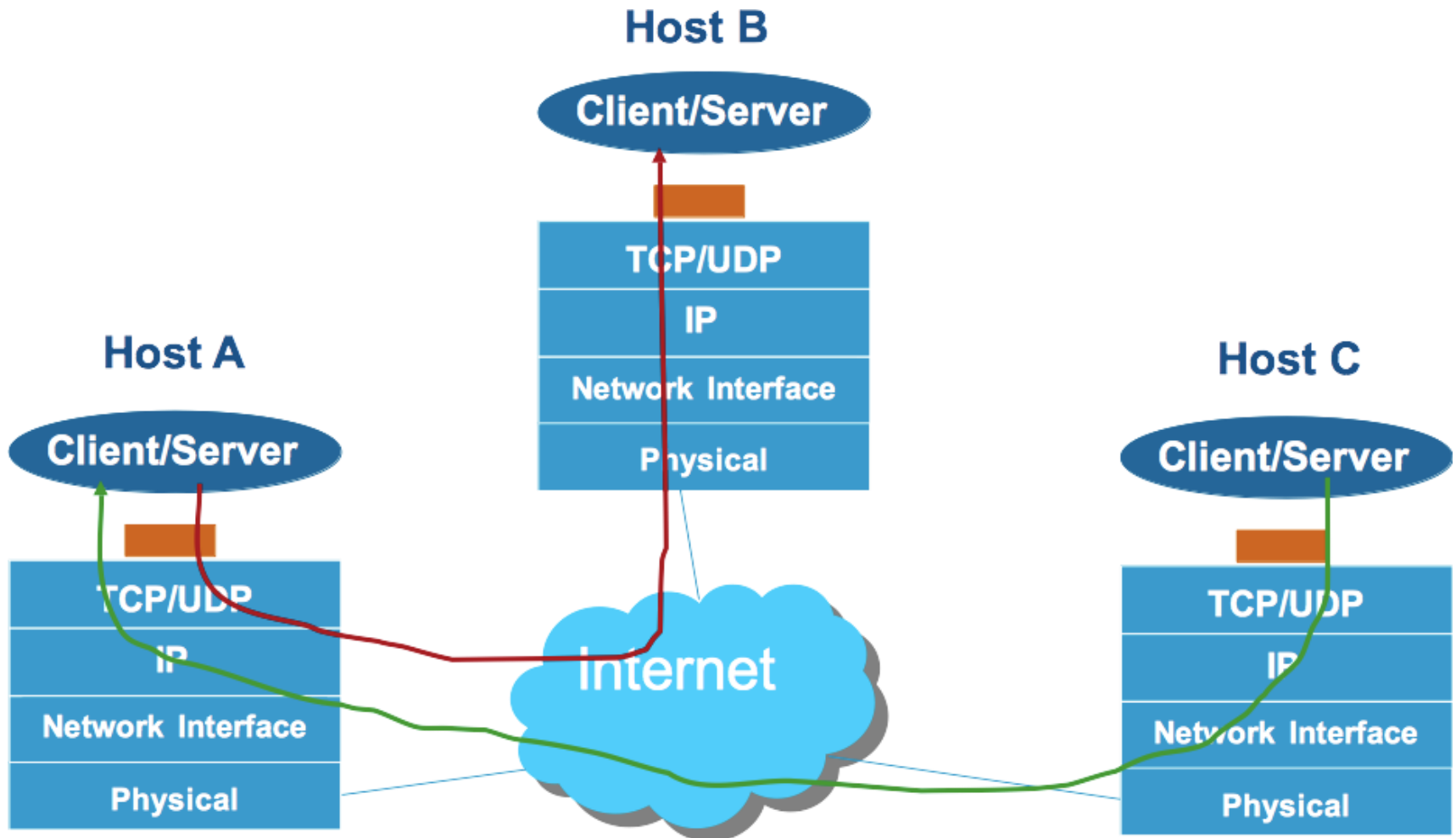
Client-Server in Sistemi Distribuiti



Client-Server Communication



Modello P2P



Remote Procedure Calls

- Remote procedure call (RPC) abstracts procedure calls between processes on networked systems
- **Stubs** – client-side proxy for the actual procedure on the server
- The client-side stub locates the server and *marshalls* the parameters
- The server-side stub receives this message, unpacks the marshalled parameters, and performs the procedure on the server

RPC: general issues

- Data Representation
 - Little endian vs. big endian
 - A system-independent representation is used (e.g., XDR: eXternal Data Representation)
- Exactly-once semantic
 - Acks and retransmissions for avoid message losses (at least once)
 - Timestamps for avoiding multiple execution (at most once)

RPC: general issues

- Client-server communication
 - How to locate the RPC port on the server?
- Predefined ports
 - RPC are associated at compile time with fixed port numbers
 - The server cannot change the port number of the required service
- Rendez-Vous
 - The server-side OS provides a rendez-vous daemon (*matchmaker*)
 - The client requires the port number to the matchmaker
 - The daemon replies with the port number
 - The client send the RPC request to the appropriate port number

An RPC Application

- Distributed File System (DFS)
 - Set of daemons and RPC clients
 - Messages containing file-system operations
 - ▶ read, write, rename, delete, status
 - The client sends a message to the server
 - The command is executed on the server
 - A reply message is sent to the client

Remote Method Invocation

- Remote Method Invocation (RMI) is a Java mechanism similar to RPCs
- RMI allows a Java program on one machine to invoke a method on a remote object

