Distributed Systems

Giuseppe Anastasi
g.anastasi@iet.unipi.it
Pervasive Computing & Networking Lab. (PerLab)
Dept. of Information Engineering, University of Pisa

Overview

- Introduction
- Network-Based Operating Systems
- Communication Networks
- Communication Protocols
- Distributed Programming

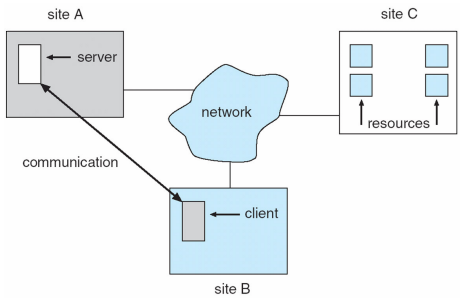
Objectives

- Provide a high-level overview of distributed systems and the networks that interconnect them
- Discuss the general structure of operating systems for distributed systems
- Introduce different types of communication networks
- Discuss networking protocols that allow communication in a distributed environment
- Introduce principles of distributed programming

Motivation

- Distributed system is collection of loosely coupled processors interconnected by a communications network
- Processors variously called *nodes, computers, machines, hosts*
  - *Site* is location of the processor
- Reasons for distributed systems
  - Resource sharing
    - sharing and printing files at remote sites
    - processing information in a distributed database
    - using remote specialized hardware devices
  - Computation speedup – load sharing
  - Reliability – detect and recover from site failure, function transfer, reintegrate failed site
  - Communication – message passing

A Distributed System



Overview

- Introduction
- Network-based Operating Systems
- Communication Networks
- Communication Protocols
- Distributed Programming

Operating Systems for Distributed Systems

- Network Operating Systems
- Distributed Operating Systems

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 (telnet, ssh)
  - Remote Desktop (Microsoft Windows)
  - Transferring data from remote machines to local machines, e.g., via File Transfer Protocol (FTP) or Web (HTTP)

Distributed-Operating Systems

- Users not aware of multiplicity of machines
  - Access to remote resources similar to access to local resources
- Data Migration
  - transfer data by transferring entire file, or only those portions of the file necessary for the immediate task
- Computation Migration
  - transfer the computation, rather than the data, across the system
- Process Migration
  - execute an entire process, or parts of it, at different sites

Distributed-Operating Systems (Cont.)

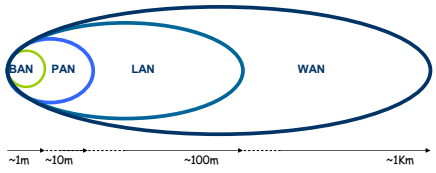
- **Load balancing**
  - distribute processes across network to even the workload
- **Computation speedup**
  - Sub-processes can run concurrently on different sites
- **Hardware preference**
  - process execution may require specialized processor
- **Software preference**
  - required software may be available at only a particular site
- **Data access**
  - run process remotely, rather than transfer all data locally

Overview

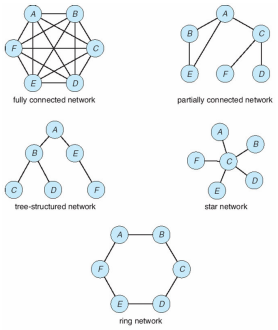
- Introduction
- Network-Based Operating Systems
- Communication Networks
  - LAN
  - WAN
  - Internet
- Communication Protocols
- Distributed Programming

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 Topologies



Service Types

- Connection Oriented (Stream)
  - Inspired from the telephone system
  - Connection Setup, Data Transfer, Connection Tear down
  - Messages tend to follow the same path from source to destination
- Connectionless (Datagram)
  - Inspired from the mail system
  - Each message includes the destination address
  - Different messages follows different paths
  - No guarantee on message ordering

Layered Architecture

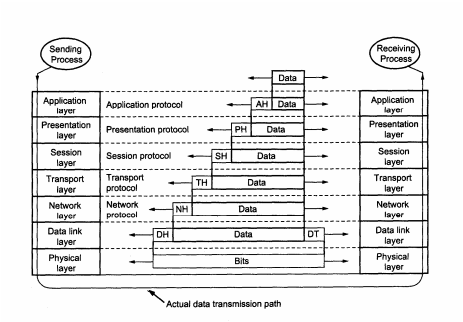
OSI

Application	LAYER 7
Presentation	LAYER 6
Session	LAYER 5
Transport	LAYER 4
Network	LAYER 3
Data Link	LAYER 2
Physical	LAYER 1

TCP/IP

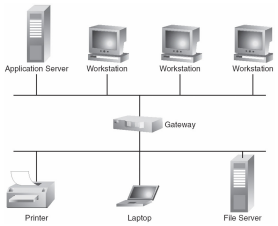
Application	LAYER 5
Transport	LAYER 4
Internet	LAYER 3
Network Interface	LAYER 2
Physical	LAYER 1

Actual Data Path



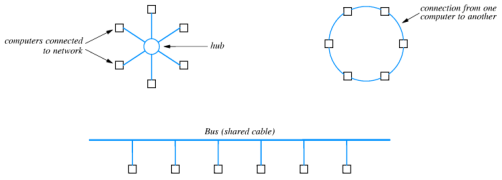
Local Area Networks (LANs)

- Cover small geographical area.
- Speed ≈ 10 – 100 Mbps
- Nodes:
  - usually workstations and/or personal computers
  - a few (usually one or two) mainframes
- Shared Communication Medium
- Broadcast communication



LAN Topologies

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 o 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.

Ethernet LANs

- Ethernet a 10 Mbps
  - 10BaseT
  - 10Base5
  - 10Base2
- Fast Ethernet (100 BaseT, 100 Mbps)
- Gigabit Ethernet (1, 10 Gbps)
  - IEEE 802.3z (1 Gbps)
  - IEEE 802.3ae (10 Gbps)

An Example of Ethernet LAN

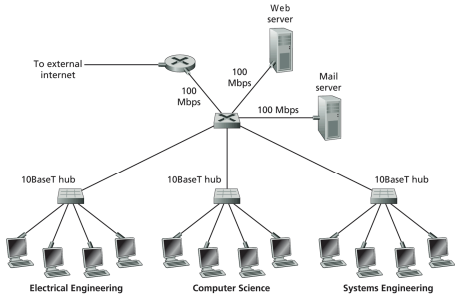
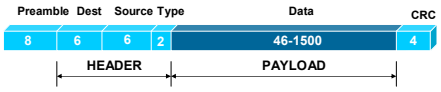


Figure 5.29 ♦ An institutional network using a combination of hubs, Ethernet switches, and a router

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

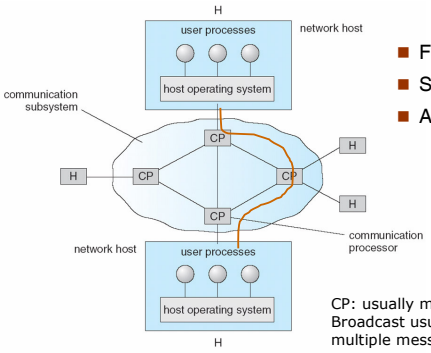
Wide Area Networks (WANs)

- Links geographically separated sites
- Point-to-point connections over long-haul lines (often leased from a phone company)
- Speed ≈ 1.544 – 45 Mbps

Point-to-Point Physical Links

- Collegamenti via modem
- Linee dedicate
- Linee ISDN
- Linee DSL
  - ADSL (Asynchronous Digital Subscriber Line)
  - SDSL (Synchronous Digital Subscriber Line)
  - HDSL (High-Rate Digital Subscriber Line)
  - VDSL (Very high-rate Digital Subscriber Line)

Point-to-Point Virtual Links

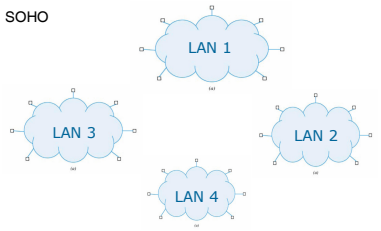


- Frame Relay
- SMDS
- ATM

CP: usually mainframes
Broadcast usually requires multiple messages

Limitazioni delle LAN

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

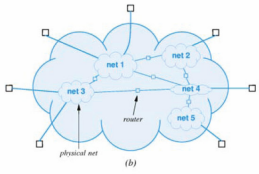


Come far comunicare fra loro calcolatori collegati a reti di tipo diverso e in posti diversi?

Quello che vorremmo …

Due calcolatori devono poter comunicare indipendentemente dalla rete fisica a cui sono collegati

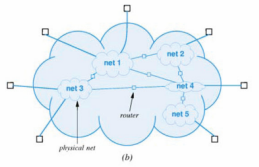
- Aumento di produttività
- Problemi da risolvere
  - Diversi segnali elettrici,
  - Diverso formato dei pacchetti
  - Diverso schema di indirizzamento



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
  - si può collegare un calcolatore come si farebbe con una rete singola
- Nascondono i dettagli delle reti fisiche sottostanti
  - L'utenti 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

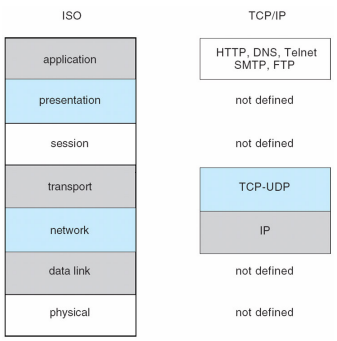
Protocolli TCP/IP

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

The TCP/IP Protocol Layers



Overview

- Introduction
- Network-Based Operating Systems
- Communication Networks
- Communication Protocols
  - TCP/UDP
  - IP
- Distributed Programming

Internet Layer (Protocollo IP)

- 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)

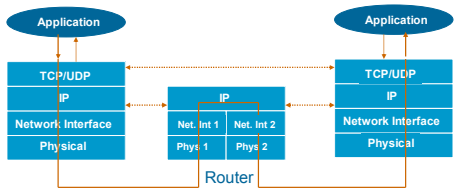
- Demultiplexing dei datagram
  - Riceve un flusso indistinto di datagrammi IP
  - Recapita i datagram ai processi applicativi a cui sono destinati
- Nessun incremento al servizio offerto da IP
  - Servizio connectionless e non affidabile

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



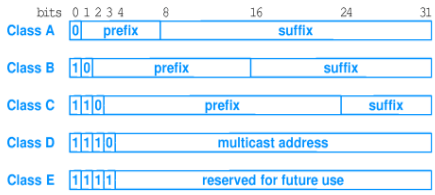
Indirizzi IP

- 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

32-bit Binary Number	Equivalent Dotted Decimal
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

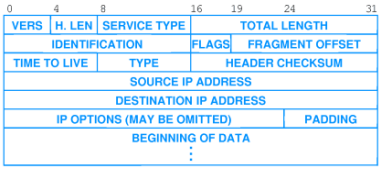
Classi e notazione puntata

Classe	Range di Valori
A	da 0 a 127
B	da 128 a 191
C	da 192 a 223
D	da 224 a 239
E	da 240 a 255

Indirizzi IP riservati

- Indirizzo per l'intera sottorete
  - Suffisso di tutti zeri (131.114.0.0)
- Indirizzo di trasmissione brodcast orientata
  - Suffisso di tutti 1 (131.114.255.255)
- Indirizzo di trasmissione brodcast 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

Datagram IP

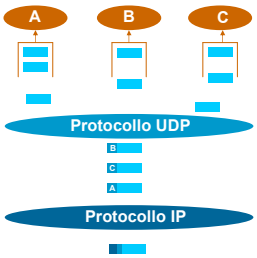


Livello di trasporto

- 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



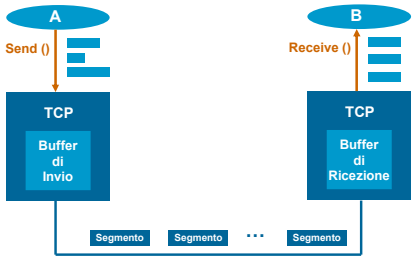
Introduzione al TCP

- 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



Overview

- Introduction
- Network-Based Operating Systems
- Communication Networks
- Communication Protocols
- Distributed Programming
  - Socket interface
  - Client-server Model
  - Peer-to-Peer Model

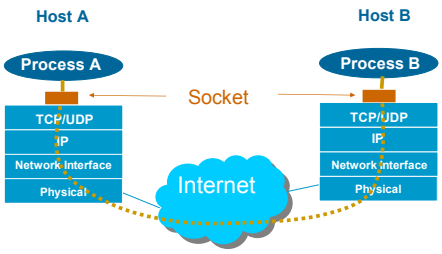
Interfaccia Socket

- 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

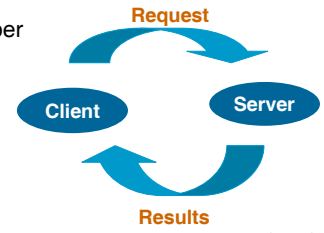


Supporto del SO

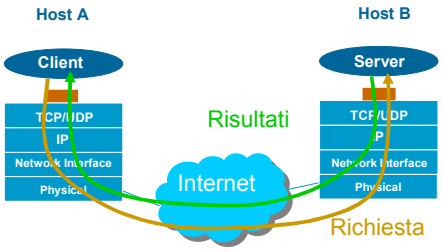
- Il SO implementa l'astrazione di socket
- System call per
  - Creare un socket
  - Associare indirizzo IP e porta al socket
  - Mettere in ascolto un processo su un socket (server)
  - Accettare una richiesta di servizio su un socket (server)
  - Aprire una connessione verso un socket remoto (client)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

Modello Client-Server

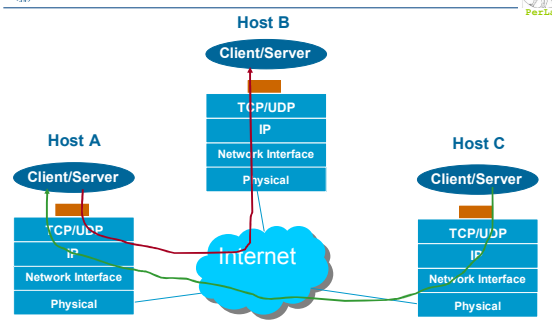
- Paradigma basato su scambio di messaggi
  - Paradigma generale
  - Ma usato principalmente in ambito distribuito
- Scambio di msg per
  - Richiesta di servizio
  - Invio dei risultati



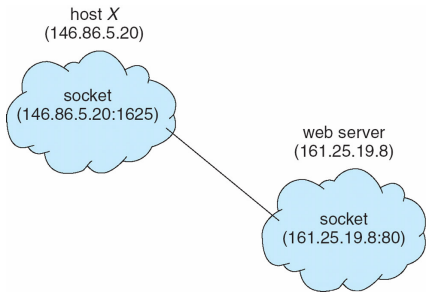
Client-Server in Sistemi Distribuiti



Modello P2P



Client-Server Communication



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

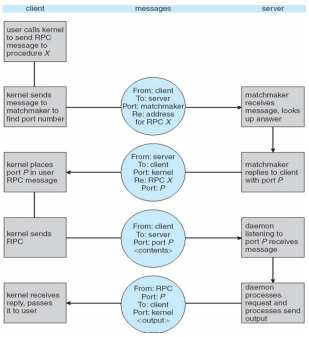
RPC: general issues

- 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

Execution of RPC

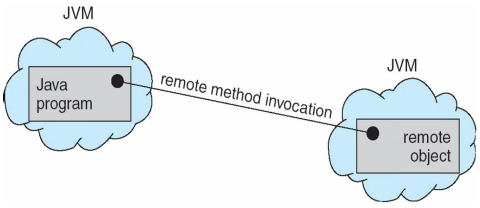


An RPC Application

- 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



Questions?