

Capitolo 5 – Parte 2 Le infrastrutture software

Architettura del SW di rete
Il Sistema Operativo di rete

I protocolli di comunicazione

- > Per comunicare i calcolatori debbono seguire delle regole: i **protocolli di comunicazione**.
- > I protocolli di comunicazione specificano:
 - i formati dei dati,
 - la struttura dei pacchetti (inclusendo la definizione delle informazioni di controllo)
 - la velocità di trasmissione
 - ...
- > Definire tutte queste proprietà tramite un unico protocollo è praticamente impossibile, per questo si definisce un **insieme di protocolli**:
 - ogni protocollo gestisce univocamente una componente ben definita della comunicazione
 - ogni protocollo condivide con gli altri protocolli i dati di cui essi necessitano.
- > Architettura di protocolli organizzata **a livelli**:
 - ogni protocollo formalizza un diverso livello di astrazione della comunicazione tra calcolatori;
 - le funzioni associate a ogni livello sono ben definite e omogenee tra loro;
 - un cambiamento nel protocollo di un livello non influenza i protocolli definiti per gli altri livelli;
 - scopo di ogni livello è di fornire servizi alle entità del livello immediatamente superiore, mascherando il modo in cui questi sono implementati e sfruttando opportunamente i servizi che gli vengono a sua volta forniti dal livello immediatamente inferiore;
 - le interfacce tra i diversi livelli sono strutturate in modo da ridurre il più possibile la necessità di scambi di informazione da un livello all'altro.

27/10/02

Introduzione ai sistemi informatici

2

Per comunicare ... /1

... è necessario che esista un **canale fisico** adatto
(requisito per la **connessione fisica**)



SSPAL - 16/04/02

ICT: un quadro tecnologico

3

Per comunicare ... /2

... è necessario avere **competenze linguistiche comuni**
(requisito per la **trasmissione**)



27/10/02

Introduzione ai sistemi informatici

4

Per comunicare ... /3

... è necessario **avere competenze di contenuto comuni**
(requisito per la **comunicazione**)



27/10/02

Introduzione ai sistemi informatici

5

Si comunica!

cosa pensi dei problemi
attuali di TCP/IP?

ritengo che IPv6 sarà
fondamentale nel futuro



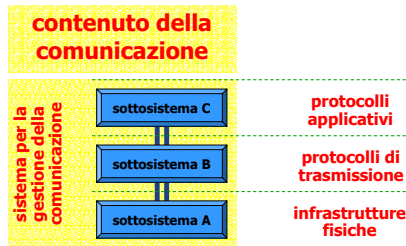
Se le soluzioni adottate per soddisfare le tre
precondizioni sono efficienti, la comunicazione si
realizza *come se* esse non fossero più necessarie
("trasparenza" dei protocolli e dell'infrastruttura)

27/10/02

Introduzione ai sistemi informatici

6

L'architettura del sistema

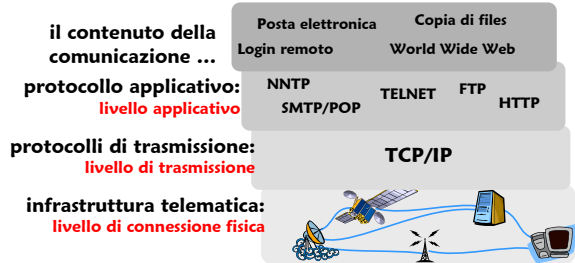


27/10/02

Introduzione ai sistemi informatici

7

La struttura di Internet



27/10/02

Introduzione ai sistemi informatici

8

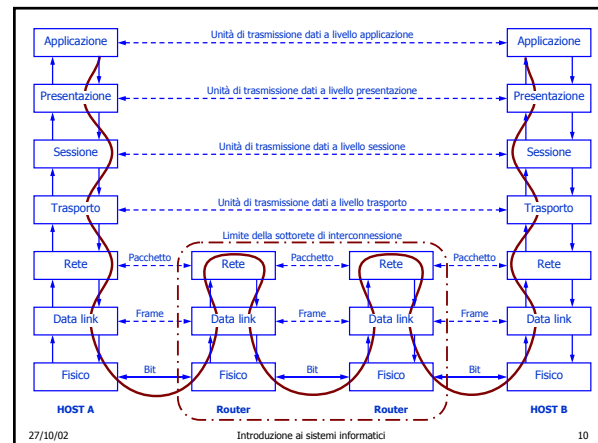
Architettura a livelli di un insieme di protocolli

- La trasmissione dell'informazione avviene simulando la connessione tra i livelli corrispondenti (**peer**) dei due sistemi che si scambiano blocchi formattati di dati, seguendo le regole stabilite dal protocollo definito per quel livello. Gli elementi chiave di un protocollo sono pertanto:
 - la sintassi da seguire per la formattazione dei blocchi dei dati;
 - la semantica, che riguarda, per esempio, le modalità di controllo della trasmissione e di gestione degli errori;
 - la temporizzazione, ovvero l'adattamento della comunicazione alla velocità di trasmissione e la sequenzializzazione delle attività.
- Modello ISO-OSI:
 - International Standard Organization (ISO)**,
 - Open Systems Interconnect (OSI)**.
 - Nel modello ISO-OSI, la comunicazione è originata dal livello più alto della stazione che invia il messaggio, passa ai livelli inferiori (sette in tutto), in cui il messaggio viene elaborato e preparato per la trasmissione, fino a giungere al livello fisico, che si occupa dell'effettiva trasmissione verso la stazione di destinazione.

27/10/02

Introduzione ai sistemi informatici

9



27/10/02

Introduzione ai sistemi informatici

10

TCP/IP

- Modello "Internet" impostato su un'architettura a cinque livelli:
 - livello fisico**
Interfaccia fisica tra le stazioni per la trasmissione dei dati e il mezzo di trasmissione.
 - livello di accesso alla rete**
Scambio dati fra un sistema finale e la rete a cui è collegato, specificando come organizzare i dati in **frame** e come trasmetterli sulla rete.
 - livello internet - IP (Internet Protocol)**
Scambio di dati tra sistemi che non appartengono alla stessa rete: occorrono delle procedure per attraversare reti multiple interconnesse. Specifica il formato dei pacchetti inviati attraverso la rete e i meccanismi utilizzati per farli transitare dal calcolatore sorgente attraverso uno o più router verso il destinatario.
 - livello di trasporto (host to host) - TCP (Transmission Control Protocol)**
Trasmissione affidabile, con la garanzia che tutte giungano a destinazione nello stesso ordine di partenza.
 - livello di applicazione**
come un'applicazione può utilizzare l'insieme dei protocolli TCP/IP.
- Il successo di questa architettura si deve alle seguenti ragioni:
 - è stata ed è un'eccellente piattaforma per la realizzazione di applicazioni client-server **affidabili ed efficienti** in particolare nell'ambito di reti geografiche;
 - ha permesso da subito di condividere informazioni tra organizzazioni diverse;
 - è stato implementato nella gran parte dei sistemi operativi ed è stato supportato da subito dai produttori di bridge e router.

27/10/02

Introduzione ai sistemi informatici

11

TCP/IP: indirizzamento

- Schema di indirizzamento generale su due livelli:
indirizzo IP + porta TCP

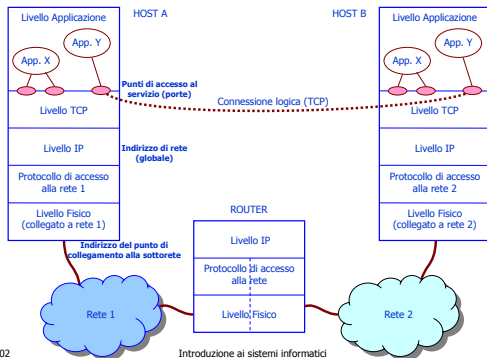
- Indirizzo IP**
 - indirizzo associato a ogni calcolatore collegato a una sottorete;
 - si tratta di un indirizzo **Internet** globale unico, utilizzato da IP per l'instradamento e la consegna dei pacchetti.
- Porta TCP**
 - indirizzo unico all'interno dell'host che individua un processo attivo sull'host;
 - utilizzato da TCP per consegnare i dati al processo giusto;
 - TCP aggiunge altre informazioni di controllo/servizio:
 - il **numero d'ordine** nella sequenza (riordinare i messaggi dopo il loro arrivo a destinazione);
 - codici di controllo della correttezza (checksum)**, che permettono al destinatario di verificare l'assenza di errori;
 - ...

27/10/02

Introduzione ai sistemi informatici

12

Indirizzi TCP/IP



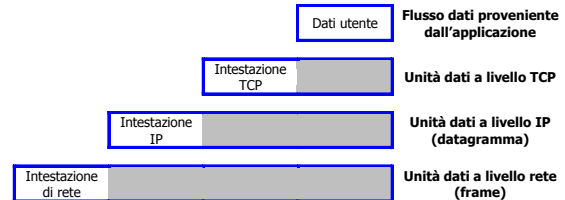
27/10/02

Introduzione ai sistemi informatici

13

Da un livello all'altro

- Ogni livello attraversato aggiunge un'intestazione (contiene informazioni utili alle funzioni proprie di quel livello):
 - **TCP** (porta TCP, checksum, numero d'ordine, ...)
 - **IP** (indirizzo host destinazione, indirizzo host mittente, ...)
 - **rete** (MAC address destinazione, MAC address mittente, ...)
 - ...



27/10/02

Introduzione ai sistemi informatici

14

Indirizzo IP (versione 4)

- 32 bit (cioè 4 byte) per un totale di 2^{32} possibili indirizzi diversi;
- rappresentato in forma "dotted decimal"
 - successione di quattro numeri (uno per byte), separati da un punto (e.g. 102.54.94.97)
 - ognuno dei quattro numeri della notazione dotted decimal è compreso tra 0 e 255.
- strutturato in due parti:
 - una parte che individua la rete fisica a cui la stazione è collegata,
 - l'altra che identifica la singola stazione nell'ambito della rete fisica;
 - esistono tre classi primarie, chiamate A, B e C, ognuna caratterizzata da una diversa suddivisione dei 32 bit:
 - A - un byte (8 bit) per la rete + 3 byte (24 bit) per i calcolatori; inizia per "0";
 - B - 2 byte (16 bit) per la rete + 2 byte (16 bit) per le stazioni; inizia per "10";
 - C - 3 byte (24 bit) per la rete + 1 byte (8 bit) per i calcolatori; inizia per "110".

27/10/02

Introduzione ai sistemi informatici

15

Affidabilità su TCP/IP

- IP è un protocollo **connectionless** (non orientato alla connessione)
 - frammenta il messaggio in datagrammi;
 - ogni datagramma viene inviato a destinazione lungo percorsi indipendenti;
 - il controllo (*checksum*) consente soltanto la verifica dell'integrità dell'intestazione, ma non dei dati;
 - attenzione:
 - non c'è garanzia che tutti i pezzi arrivino a destinazione né che arrivino "in ordine"
 - la correttezza e l'ordine di ricezione dei dati devono essere assicurati da protocolli di livello più elevato.
- TCP è un protocollo **connection oriented** (orientato alla connessione)
 - garantisce la consegna di un messaggio completo di tutte le sue parti e ordinato correttamente,
 - il controllo consente la valutazione della correttezza sia dell'intestazione TCP che dei dati.
- La combinazione delle due modalità permette di ottenere sia una buona efficienza di trasmissione che una elevata affidabilità:
 - OK per applicazioni **client-server**;
 - KO laddove l'affidabilità non è un requisito essenziale. In questi casi TCP può essere sostituito con altri protocolli (e.g. UDP - User Datagram Protocol).

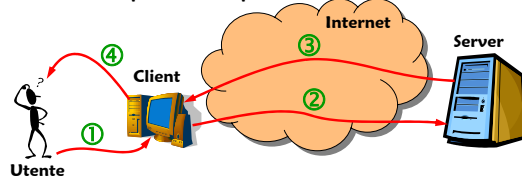
27/10/02

Introduzione ai sistemi informatici

16

Il paradigma client-server

1. L'utente usa il client per esprimere le sue richieste
2. Il client si collega al server e trasmette la richiesta
3. Il server risponde al client
4. Il client presenta la risposta all'utente



27/10/02

Introduzione ai sistemi informatici

17

Il client



- Si preoccupa di dialogare con l'utente
- Sfrutta tutte le possibilità fornite dal calcolatore su cui viene eseguito (audio, video, ...)
- Fornisce all'utente un'interfaccia intuitiva
- Elabora le richieste dell'utente e le risposte dei server
 - la comunicazione avviene secondo un formato standard (protocollo)

27/10/02

Introduzione ai sistemi informatici

18

Il server



- Rende disponibili delle risorse
- Accetta richieste e risponde automaticamente
 - non bada alla provenienza della richiesta
 - il processo client può trovarsi in qualsiasi punto della rete
- Si può organizzare un insieme di server in modo che siano collegati tra loro
- Potrebbe essere eseguito dallo stesso calcolatore che esegue il processo client!

27/10/02

Introduzione ai sistemi informatici

19

Indirizzi numerici vs indirizzi simbolici

- Gli indirizzi IP sono **machine-oriented**, quindi difficili da utilizzare per un utente "umano";
- è stato definito un sistema per passare da indirizzi numerici (gli **indirizzi IP**) a **nomi** facilmente memorizzabili, il **Domain Name System**;
- **Domain Name System (DNS)**
 - associa a ogni indirizzo IP uno o più indirizzi simbolici,
 - gestisce la conversione tra indirizzi simbolici e indirizzi IP
- organizzato in **maniera gerarchica** (domini, sotto-domini, sotto-sotto-domini, ...) per semplificarne l'utilizzo.



27/10/02

Introduzione ai sistemi informatici

20

DNS

- Il nome DNS di un calcolatore è costituito da una successione di stringhe alfanumeriche separate da punti (per esempio, server1.isttec.liuc.it)
- ogni stringa identifica un "dominio":
 - la stringa più a destra rappresenta il dominio di primo livello (detto anche dominio generale)
 - identifica la nazione di appartenenza (it per Italia, uk per Gran Bretagna, fr per Francia...)
 - identifica la categoria cui appartiene la società proprietaria del calcolatore (com per commerciale, edu per università o istituzioni che si occupano di formazione, org per organizzazioni non-profit di vario genere...);
 - la seconda stringa, sempre proseguendo da destra verso sinistra, indica il dominio di secondo livello
 - è un sottodominio del dominio di primo livello
 - di solito individua una singola organizzazione (università, azienda, ente...)
 - Le stringhe successive indicano i domini di terzo livello (sottodomini dei domini di secondo livello), quelli di quarto livello, e così via finché non si arriva a individuare un dominio che comprende il singolo host.

27/10/02

Introduzione ai sistemi informatici

21

Come si passa da DNS a IP?

- A ogni dominio è associato a un **calcolatore responsabile del dominio**
 - si consideri, per esempio, l'indirizzo **server1.isttec.liuc.it**:
 - c'è un computer responsabile per il dominio **it**;
 - un computer per il dominio **liuc.it**;
 - un terzo computer per il dominio **isttec.liuc.it**;
 - un ulteriore computer per **server1.isttec.liuc.it**.
- Il calcolatore responsabile di un dominio mantiene un **elenco dei calcolatori responsabili dei suoi sottodomini** (e ne conosce i relativi indirizzi IP),
 - il calcolatore responsabile del dominio **it**, per esempio, deve sapere chi sono (cioè deve sapere qual è il loro indirizzo IP) i calcolatori responsabili di tutti i suoi sottodomini, tra cui c'è **liuc.it**, ma anche **polimi.it**, **miur.it**...
 - il calcolatore responsabile del dominio **liuc.it**, per esempio, deve sapere chi sono i calcolatori responsabili di tutti i suoi sottodomini, tra cui c'è **isttec.liuc.it**, ma anche **ctic.liuc.it**, ...
- Per tradurre l'indirizzo DNS di un calcolatore nel suo indirizzo IP si deve **interrogare il responsabile di ciascuno dei domini** (di I, II, ... livello) cui quel calcolatore appartiene:
 - il calcolatore responsabile del dominio di I livello sa dove si trova il calcolatore responsabile del dominio di II livello
 - il calcolatore responsabile del dominio di II livello sa dove si trova il calcolatore responsabile del dominio di III livello
 - ...

27/10/02

Introduzione ai sistemi informatici

22

Il SO nelle reti di calcolatori

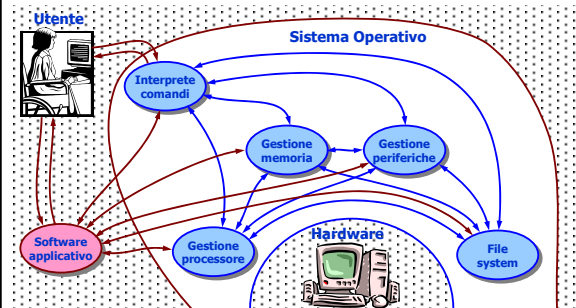
- Le reti (specie quelle locali – LAN) sono molto veloci, quindi è possibile estendere anche a programmi in esecuzione su calcolatori diversi il principio di virtualizzazione delle risorse:
 - si perde il concetto di "localizzazione" delle risorse
 - c'è la possibilità di condividere in modo trasparente dati, periferiche e unità di elaborazione.
- Tali principi si possono estendere verso collegamenti (reti geografiche – WAN) su scala nazionale o internazionale:
 - è possibile per esempio fornire a un utente (sia esso un programma o un utente umano) una visione *unificata* dei dati relativi a un sistema bancario composto da molte filiali, in cui ognuna memorizza i propri dati su calcolatori diversi, distanti, ma collegati tramite una rete di comunicazione.
- Per i calcolatori collegati in rete, il **sistema operativo** deve gestire anche quelle **risorse** che sono **accessibili tramite la rete stessa**.

27/10/02

Introduzione ai sistemi informatici

23

Elementi di un SO

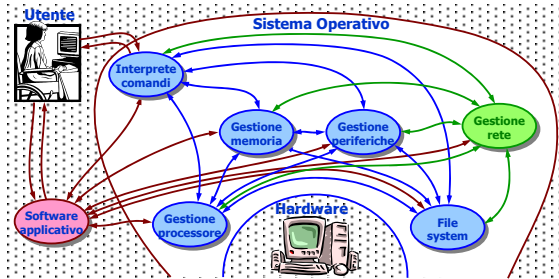


27/10/02

Introduzione ai sistemi informatici

24

Elementi di un SO di rete



27/10/02

Introduzione ai sistemi informatici

25

NOS vs DOS

- Si distinguono due categorie di SO che gestiscono anche risorse di rete:
 - network operating systems, NOS**
 - approccio tradizionale: le funzionalità di gestione della rete sono aggiunte al sistema operativo locale
 - il gestore della rete agisce da filtro tra il sistema locale e i sistemi remoti;
 - distributed operating systems, DOS**
 - esercitano effettivamente un controllo su scala globale delle risorse
 - l'accesso alle risorse è globale, senza una simulazione di accesso locale.

	NOS	DOS
Controllo delle risorse	Nodi locali	Sistema globale
Gestione risorse locali	Nodo locale	DOS
Controllo di accesso	Locale	DOS
Gestione delle richieste	Passate da un sistema locale all'altro tramite il NOS	Passano da un nodo all'altro tramite DOS

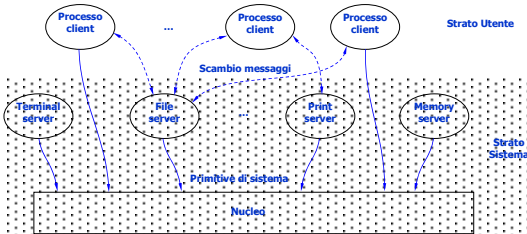
27/10/02

Introduzione ai sistemi informatici

26

Organizzazione client-server

- Obiettivo: **minimizzare le dimensioni del nucleo**
 - si spostano alcune componenti del sistema verso gli **strati applicativi**
 - le funzionalità estranee al nucleo sono "servizi" forniti da **processi server**.



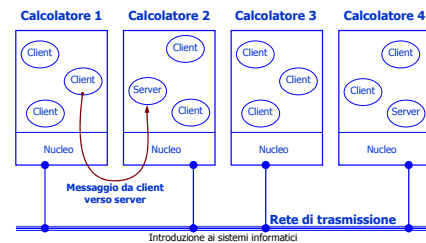
27/10/02

Introduzione ai sistemi informatici

27

Organizzazione client-server

- I **client** possono usufruire dei servizi di sistema inviando una richiesta al **server**
 - il nucleo deve gestire le comunicazioni tra i client e i server;
 - il nucleo deve gestire i processi e la concorrenza tra processi;
 - i processi sono isolati, ovvero non condividono dati se non attraverso il nucleo.
- Ottime proprietà di **modularità** e **portabilità**:
 - le politiche di gestione delle risorse sono demandate interamente ai processi server, che sono sviluppati in modo indipendente dalle funzionalità del nucleo che utilizzano;
 - le funzioni per la gestione delle risorse non dipendono dalle diverse possibili realizzazioni delle risorse stesse, quindi uno stesso sistema operativo è portabile su calcolatori diversi.



27/10/02

Introduzione ai sistemi informatici

28

Client-server

- Vantaggi di questa strutturazione:
 - indipendenza fisica fra le applicazioni client e applicazioni server;
 - possibilità di specializzare l'interfaccia utente del client lasciando invariato l'insieme dei servizi del server;
 - possibilità di aggiornare tecnologicamente e funzionalmente il server in modo trasparente ai client;
 - riduzione degli interventi di manutenzione del software e garanzia di coerenza degli aggiornamenti.
- Diffusione destinata a rafforzarsi ulteriormente:
 - introduzione di palmari wireless: client in grado di connettersi a processi server accessibili tramite la rete.
- Microkernel:** struttura moderna in cui si assegnano al kernel poche funzioni essenziali
 - gestione degli spazi di indirizzamento,
 - comunicazione fra i processi (**IPC, Inter-Process Communication**)
 - funzioni base di gestione della CPU (scheduling).
 - i servizi del SO sono forniti da processi che sono trattati dal microkernel come tutte le altre applicazioni.

27/10/02

Introduzione ai sistemi informatici

29

I modelli ibridi

- Organizzazione ibrida: coesistono paradigmi di interazione basati su dati condivisi (come nel modello a strati) con altri che impiegano la comunicazione di messaggi (come nel modello client-server).
 - i processi (**thread**) sono suddivisi in gruppi,
 - i componenti dello stesso gruppo condividono i dati,
 - l'interazione con thread di altri gruppi avviene tramite messaggi.
- Multi-threading** è utile per quelle applicazioni che svolgono compiti ragionevolmente indipendenti che non necessitano pertanto di essere serializzati;
 - per esempio un server di un database che elabora le richieste di diversi client associate a dei thread
 - il passaggio da un thread ad un altro dello stesso gruppo richiede un sovraccarico di commutazione molto minore rispetto al passaggio da un processo a un altro.
- Microkernel** e **multithreading** sono ormai usati in maniera congiunta nello sviluppo dei sistemi operativi e delle applicazioni distribuite.

27/10/02

Introduzione ai sistemi informatici

30

Bytecode: un modello per la distribuzione di programmi in rete

- Lo sviluppo di un programma distribuito via web segue una progressione leggermente diversa da quella tradizionale:
 - **scrivere** il programma (con un normale editor o in un ambiente integrato, **Integrated Development Environment, IDE**),
 - **compilare** il sorgente per ottenere un formato intermedio, detto **bytecode**, simile al linguaggio macchina ma compreso da un interprete;
 - **caricare** il programma, in formato bytecode, in memoria a partire da un disco o **direttamente dalla rete**;
 - **verificare** che il programma rispetti le norme di sicurezza;
 - **interpretare** il bytecode.
- Vantaggi:
 - rendere le applicazioni indipendenti dalla piattaforma di esecuzione;
 - incrementare l'efficienza rispetto all'interprete tradizionale, poiché alcune ottimizzazioni tipiche dei compilatori possono essere attuate anche nella generazione del bytecode.

27/10/02

Introduzione ai sistemi informatici

31

La gestione dei processi

- Rete di calcolatori = generalizzazione di un sistema multiprocessore
 - sono disponibili diversi esecutori, almeno uno per ogni nodo della rete
 - i processi che possono trovarsi nello stato di esecuzione sono più di uno, in genere uno per ogni processore presente nel sistema.
- Gestione dei processi:
 - definire come gestire l'evoluzione di un singolo processo attraverso i possibili stati di esecuzione, attesa e pronto;
 - definire come distribuire tra i diversi nodi (o meglio tra i diversi processori) i processi che sono attivi nel sistema.
- Per gestire correttamente questi due aspetti, il sistema di gestione dei processi è articolato in due parti:
 - la prima compresa nel nucleo del sistema operativo (detto **kernel**) installato in ciascuno dei nodi della rete,
 - la seconda compresa in un modulo di coordinamento globale (detto **dispatcher**).

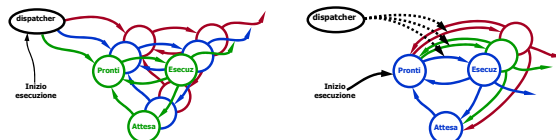
27/10/02

Introduzione ai sistemi informatici

32

Dispatcher

- Decide come distribuire i processi tra i diversi nodi (e quindi i diversi kernel) secondo alcuni criteri:
 - distribuire il carico tra i diversi processori (evitare il sovraccarico);
 - mantenere la località del processo (evitare ritardi dovuti alla trasmissione dei dati lungo la rete).
- Le operazioni di scheduling del dispatcher hanno un orizzonte temporale più ampio rispetto alle decisioni dei kernel.



27/10/02

Introduzione ai sistemi informatici

33

La gestione della memoria

- La memoria è distribuita, in maniera più o meno uniforme, sui diversi nodi della rete di calcolatori.
 - il tempo di risposta del sistema di memoria varia notevolmente a seconda che il dato richiesto sia presente nella memoria locale piuttosto che in una memoria remota
 - per migliorare le prestazioni la memoria viene allocata secondo il principio di località: mantenere sullo stesso nodo il processo e la memoria che gli è stata assegnata.
- Gestione dei messaggi:
 - il sistema operativo (distribuito o di rete) fornisce dei servizi che permettono di spedire (**send**) o di ricevere (**receive**) un messaggio;
 - mittente e destinatario dei messaggi possono essere eseguiti su nodi diversi.

27/10/02

Introduzione ai sistemi informatici

34

La gestione delle periferiche

- Il principio di località non prevale sulla necessità di rendere un dispositivo accessibile da ogni nodo della rete (al contrario di quanto succede per la memoria).
- Una periferica collegata a un nodo della rete debba essere utilizzata anche da processi eseguiti su altri nodi della rete.
- Gestore delle periferiche:
 - assegna un nome univoco a tutti i dispositivi; questo nome viene utilizzato dai diversi processi utente per accedere al dispositivo;
 - il funzionamento della periferica è controllato da un processo (detto amministratore della periferica) che ha la responsabilità di amministrarla;
 - i processi che vogliono utilizzare la periferica si rivolgono al relativo "amministratore".
- Esempio: condivisione di una stampante di rete
 - il processo che vuole stampare un documento invia i dati alla stampante
 - i dati vengono salvati come file nella cartella di **spool** del calcolatore (**print server**) cui la stampante è collegata;
 - viene notificata la presenza dei file al processo **spooler**, che controlla la stampante;
 - lo spooler accoda i file all'elenco dei file da stampare e, non appena la stampante si libera dei lavori precedenti, provvede a inviarli alla periferica.

27/10/02

Introduzione ai sistemi informatici

35

File system

- La visione della localizzazione dei file deve essere resa il più possibile indipendente dai dettagli fisici:
 - quando un utente, dopo essersi identificato, si collega al sistema, si trova a operare all'interno di una porzione del sistema complessivo che potrebbe avere file e directory memorizzati su più unità, non necessariamente locali al calcolatore da cui l'utente accede, ma che sarà utilizzata dall'utente come se fosse la radice (**home**) del suo file system personale.
- Un sistema operativo che consente una gestione distribuita del file system:
 - deve integrare in modo organico i singoli file system dei calcolatori della rete;
 - deve risolvere i problemi dell'univocità dei nomi di file e directory per i calcolatori della rete;
 - deve consentire un accesso efficiente anche a file presenti su calcolatori remoti.
- Questi requisiti vengono in genere soddisfatti utilizzando un file system di tipo client-server:
 - ogni richiesta di accesso alla memoria di massa viene effettuata verso un processo, detto **directory server**, residente su un calcolatore che funge da server per l'intero sistema;
 - il directory server crea una corrispondenza tra ogni filename e la locazione del corrispondente file, al fine di identificare ogni calcolatore nella rete e ogni file all'interno della sua memoria di massa.

27/10/02

Introduzione ai sistemi informatici

36

Il file system di rete (NFS)

- Obiettivo: evitare il più possibile il trasferimento dei file.
- TCP/IP offre un servizio di accesso ai file in rete noto come NFS (**Network File System**) che consente alle applicazioni di accedere a un file operando a distanza con una organizzazione client-server:
 - il client NFS invia la richiesta di operazione al server NFS interessato,
 - il server NFS procede alla modifica del file,
 - il server conferma al client l'avvenuta operazione,
 - in questo modo solo i dati da leggere e scrivere attraversano la rete.
- Per evitare interferenze negli aggiornamenti dei file, NFS consente a un client di richiedere un **accesso condiviso** oppure anche un **accesso esclusivo** al file.

27/10/02

Introduzione ai sistemi informatici

37

L'affidabilità del file system

- Le prestazioni dei dischi crescono più lentamente che quelle delle CPU
 - accesso ai dischi migliorato di **5/10 volte** in 20 anni, frequenza di clock delle CPU raddoppia ogni 18 mesi;
 - **gap** di prestazioni sempre **più ampio**.
- **Parallelizzazione** per migliorare le **prestazioni**
 - **RAID - Redundant Array of Inexpensive Disks** vs. **SLED - Single Large Expensive Disk**.
 - RAID = scatola piena di dischi
 - server di grosse dimensioni + controllore RAID
 - dal punto di vista del sistema operativo un RAID assomiglia ad uno SLED, ma fornisce prestazioni migliori e più affidabilità
 - realizzato in genere usando dischi SCSI
 - i dati vengono distribuiti fra le diverse unità, permettendo così il funzionamento in parallelo.

27/10/02

Introduzione ai sistemi informatici

38

RAID livello 0 (striping without parity)

- Il disco singolo virtuale simulato dal RAID è diviso in strisce di k settori ciascuna:
 - i settori da 0 a k - 1 costituiscono la striscia (**strip**) 0
 - i settori da k a 2k - 1 costituiscono la striscia 1
 - ...
- La distribuzione dei dati su unità multiple si chiama **striping**.
- Diminuisce l'affidabilità.
- Cresce la velocità se le operazioni sfruttano il parallelismo.

Strip 0	Strip 1	Strip 2	Strip 3
Strip 4	Strip 5	Strip 6	Strip 7
Strip 8	Strip 9	Strip 10	Strip 11

27/10/02

Introduzione ai sistemi informatici

39

RAID livello 1 (mirroring)

- Tutti i dischi sono **duplicati**
 - con quattro dischi principali ci sono anche quattro dischi di backup;
 - ogni scrittura viene eseguita due volte, la lettura può essere eseguita su una delle due copie:
 - le prestazioni in scrittura sono **uguali** a quelle di un'unità singola,
 - le prestazioni di lettura possono essere fino a **due volte** superiori.
- La **tolleranza agli errori** è eccezionale
 - se un'unità smette di funzionare basta usare la copia;
 - per la riparazione è sufficiente installare una nuova unità e copiarvi i dati di backup.

Strip 0	Strip 1	Strip 2	Strip 3	Strip 0	Strip 1	Strip 2	Strip 3
Strip 4	Strip 5	Strip 6	Strip 7	Strip 4	Strip 5	Strip 6	Strip 7
Strip 8	Strip 9	Strip 10	Strip 11	Strip 8	Strip 9	Strip 10	Strip 11

27/10/02

Introduzione ai sistemi informatici

40

RAID livello 2 (Hamming)

- Attivo a livello di parole o di byte.
 - Esempio: dati divisi in pezzi di 4 bit ciascuno;
 - ai 4 bit si aggiunge un codice Hamming e si forma una parola di 7 bit, dove i bit 1, 2 e 4 sono bit di parità;
 - è possibile scrivere la parola di 7 bit codificata con il codice Hamming su 7 unità, un bit per ogni unità.

Bit 0	Bit 1	Bit 2	Bit 3	Bit 4	Bit 5	Bit 6

27/10/02

Introduzione ai sistemi informatici

41

RAID livello 3

- Versione semplificata del RAID livello 2
 - per ogni parola di dati viene calcolato un bit di parità che viene scritto in un'apposita unità;
 - nota la posizione dell'errore, la parità ne consente la correzione.

Bit 0	Bit 1	Bit 2	Bit 3	Parity

27/10/02

Introduzione ai sistemi informatici

42

RAID livello 4

- Lavora con strisce, e non richiede unità sincronizzate:
 - è come il RAID livello 0 con parità associata a ciascuna striscia e scritta su un'unità supplementare
 - se ogni striscia è lunga k byte, ad esempio, si può fare l'EXCLUSIVE OR di tutte le strisce ottenendo una striscia di parità lunga k byte
 - se un'unità non funziona i byte persi si possono ricalcolare dall'unità di parità.

Strip 0	Strip 1	Strip 2	Strip 3	P 0-3
Strip 4	Strip 5	Strip 6	Strip 7	P 4-7
Strip 8	Strip 9	Strip 10	Strip 11	P 8-11
Strip 12	Strip 13	Strip 14	Strip 15	P 12-15

27/10/02

Introduzione ai sistemi informatici

43

RAID livello 5 (striping with parity)

- Lavora con strisce, e non richiede che le unità siano sincronizzate.
- Si distribuiscono le informazioni di parità su diverse unità per aumentare il parallelismo degli accessi.

Strip 0	Strip 1	Strip 2	Strip 3	P 0-3
Strip 4	Strip 5	Strip 6	P 4-7	Strip 7
Strip 8	Strip 9	P 8-11	Strip 10	Strip 11
Strip 12	P 12-15	Strip 13	Strip 14	Strip 15
P 16-19	Strip 16	Strip 17	Strip 18	Strip 19

27/10/02

Introduzione ai sistemi informatici

44