



Sistemi Distribuiti

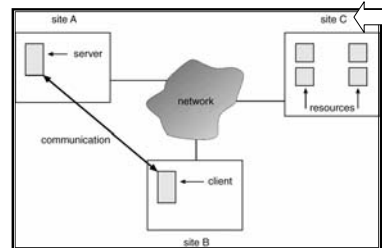
Struttura dei sistemi distribuiti

Donatella Granata 0521000050
Armando Pagliara 0521000062

Sistemi per l'Elaborazione delle Informazioni

Introduzione (1)

➤ Che cos'è un sistema distribuito?
 Un sistema distribuito è un insieme di unità di elaborazione debolmente connesse tramite una rete di comunicazione



Bibliografia: Silbershatz, Galvin, Gagne – Sistemi Operativi, concetti ed esempi Sistemi per l'Elaborazione delle Informazioni

Introduzione (2)

- Ogni unità di elaborazione in un sistema distribuito considera remote le altre unità di elaborazioni e loro risorse.
- In genere, un calcolatore (server) mette a disposizione un risorsa ad un calcolatore di un altro sito(client).
- **Scopo:**
 Fornire un ambiente efficiente e conveniente alla condivisione delle risorse

Bibliografia: Silbershatz, Galvin, Gagne – Sistemi Operativi, concetti ed esempi Sistemi per l'Elaborazione delle Informazioni

Vantaggi (1)

Condivisione delle risorse

possibilità di usare le risorse disponibili su un altro sito:

- ✓ Condividere e stampare file in un sito remoto
- ✓ Elaborazione delle informazioni in una base di dati distribuiti
- ✓ Usare dispositivi hardware specializzati

Accelerazione dei calcoli:

suddividere i calcoli in sottocalcoli processati tra diversi siti, ed eventualmente anche spostare il carico computazionale ad una macchina con carico inferiore, in modo da avere anche una condivisione di calcolo

Bibliografia: Silbershatz, Galvin, Gagne – Sistemi Operativi, concetti ed esempi Sistemi per l'Elaborazione delle Informazioni

Vantaggi (2)

Affidabilità

- Se il sistema distribuito è composto da grandi sistemi di calcolo autonomi, il malfunzionamento di un sito viene compensato dai restanti
- Per piccoli sistemi invece per evitare il collasso, si utilizza un elevato livello di ridondanza sia dei dispositivi che dei dati.

Comunicazione:

- Utenti di diversi siti possono scambiarsi informazioni.
- Con i sistemi operativi progettati come insiemi di processi comunicanti tramite i messaggi, è possibile maggiore flessibilità nella dislocazione delle risorse, per effettuare semplici manutenzioni e più servizi : RPC, posta elettronica, trasferimento dei file, sessioni di lavoro remote.

Bibliografia: Silbershatz, Galvin, Gagne – Sistemi Operativi, concetti ed esempi Sistemi per l'Elaborazione delle Informazioni

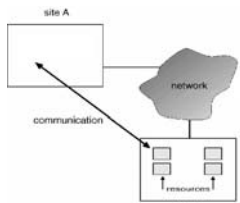
Sistemi operativi distribuiti (1)

Il sistema operativo distribuito deve garantire che l'accesso alle risorse dislocate su siti remoti sia il più simile possibile ad un accesso locale per un qualsiasi utente.

Migrazione dei dati:

Se A vuole accedere a dati che risiedono in B, 2 tipi di trasferimento:

- Tutto il file
 (considerato come sistema FTP automatico)
 Svantaggi:
 Inefficiente per file di dimensioni elevate che devono essere rispettati ad ogni modifica
- Solo la parte interessata
 Adoperato dal protocollo
 NFS(Network file system) della Sun Microsystems



Bibliografia: Silbershatz, Galvin, Gagne – Sistemi Operativi, concetti ed esempi Sistemi per l'Elaborazione delle Informazioni

Sistemi operativi distribuiti (2)

Migrazione delle computazioni:

Se un lavoro di elaborazione necessita di file di grandi dimensioni dislocati sulla rete si preferisce riportare solo i risultati al sito che effettua il calcolo invece di trasferire l'intero file

Un processo P per accedere ad un file sul sito A può:

1. Avviare una RPC predefinita che riporta i risultati a P tramite messaggi, usando il protocollo per datagrammi (UDP)
2. Inviare un messaggio al sito A, che crea un processo specifico Q, che invia i risultati una volta computati a P

Bibliografia: Silberhatz, Galvin, Gagne – Sistemi Operativi, concetti ed esempi

Sistemi per l'Elaborazione delle Informazioni

Sistemi operativi distribuiti (3)

Migrazione dei processi:

Un processo o parti di esso possono essere eseguiti su siti diversi

Vantaggi :

- **Bilanciamento del carico**
- **Accelerazione dei calcoli**
calcolo diviso tra sottoprocessi eseguiti su macchine anche su siti diversi per minimizzare i tempi
- **Preferenza di sistemi e dispositivi**
caratteristiche peculiari di un processo possono rendere più adatta l'esecuzione su una unità di elaborazione specializzata
- **Preferenza di programmi**
necessità di un programma presente su un sito
- **Accesso ai dati**
accesso remoto ai dati è preferibile ad una migrazione dei dati

Bibliografia: Silberhatz, Galvin, Gagne – Sistemi Operativi, concetti ed esempi

Sistemi per l'Elaborazione delle Informazioni

Sistemi operativi distribuiti (4)

Metodi per spostare i processi nella rete:

- Si nasconde che il processo sia un processo client

Vantaggi:

- ✓ Non si deve codificare esplicitamente il suo programma
- ✓ Bilanciamento del carico
- ✓ Accelerazione dei calcoli tra sistemi omogenei

- Richiedere all'utente come il processo deve migrare

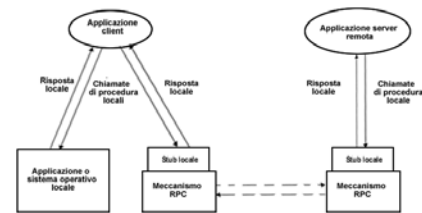
Nei casi in cui si deve trasferire un processo per soddisfare preferenze di dispositivi

Bibliografia: Silberhatz, Galvin, Gagne – Sistemi Operativi, concetti ed esempi

Sistemi per l'Elaborazione delle Informazioni

Chiamata di procedura remota (1)

- Permette a programmi su macchine diverse di comunicare mediante semantiche di chiamata/ritorno di procedure



Bibliografia: Stallings – Sistemi Operativi

Sistemi per l'Elaborazione delle Informazioni

Chiamata di procedura remota (2)

Vantaggi:

- La chiamata a procedura è una astrazione ampiamente usata
- Permette di specificare le interfacce remote come un insieme di operazioni con nomi e tipi assegnati, con controlli statici di tipo sui programmi distribuiti
- Generazione automatica mediante l'utilizzo di una interfaccia standardizzata
- Grazie ad una definizione precisa dell'interfaccia, la scrittura dei moduli per i client e i server sono dotati di una alta portabilità ovviamente senza nessuna necessità di ricodificare ed effettuare correzioni

Bibliografia: Stallings – Sistemi Operativi

Sistemi per l'Elaborazione delle Informazioni

Chiamata di procedura remota (3)

Il programma chiamante effettua una normale chiamata di procedura con i parametri opportuni sulla sua macchina

CALL P(X,Y)

P = nome della procedura

X = argomenti passati

Y = valori di ritorno

Quando si ha la necessità di invocare una RPC, una procedura fantoccio o stub P deve essere inclusa nello spazio di indirizzamento del chiamante o essere collegata al tempo della chiamata

1. Lo stub invia un messaggio al sistema remoto che identifica la procedura chiamata e ne include i parametri
2. Attende risposta
3. Restituisce i valori richiesti al programma chiamante

Bibliografia: Stallings – Sistemi Operativi

Sistemi per l'Elaborazione delle Informazioni

Chiamata di procedura remota (4)

Nella macchina remota viene associata *stub* alla procedura chiamata :

1. All'arrivo del messaggio, viene esaminato e viene generata una CALL locale
2. Viene creato i messaggi di risposta con i risultati
3. Il messaggio viene inviato



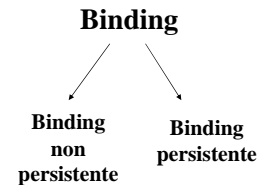
Bibliografia: Stallings - Sistemi Operativi

Sistemi per l'Elaborazione delle Informazioni

Binding client/server (1)

Il **binding** (legame) specifica come stabilire una relazione tra una procedura remota e il programma chiamante.

- Si forma ogni volta che due applicazioni hanno una connessione logica e si preparano a scambiare comandi e dati



Bibliografia: Stallings - Sistemi Operativi

Sistemi per l'Elaborazione delle Informazioni

Binding client/server (2)

➤ Binding non persistente

La connessione viene stabilita tra i due processi al momento della chiamata di procedura remota e smantellata al momento della restituzione dei risultati

Svantaggi:

- ✓ Overhead causato dalla necessità di ristabilire le connessioni rende il binding inefficiente
- ✓ Procedure chiamate frequentemente

➤ Binding persistente

La connessione viene mantenuta anche dopo la fine della chiamata a procedura remota per essere riutilizzata

- ✓ Terminazione automatica della connessione a tempi stabiliti
- ✓ Utile per sequenze di chiamate che sfruttano la stessa connessione

Bibliografia: Stallings - Sistemi Operativi

Sistemi per l'Elaborazione delle Informazioni

Confronto tra chiamate di procedure sincrone e asincrone (1)

➤ RPC sincrone

Il processo chiamante attende finché il processo chiamato non restituisce valore

- ✓ Facile da capire e programmare
- ✓ Non sfrutta appieno il parallelismo di un sistema distribuito ovviamente con prestazioni minori

➤ RPC asincrona

Non blocca il chiamante, esempio tipico è di permettere ad un cliente di invocare ripetutamente un server in modo da avere un certo numero di richieste per volta nella pipeline ognuna con il suo sistema di dati

Bibliografia: Stallings - Sistemi Operativi

Sistemi per l'Elaborazione delle Informazioni

Confronto tra chiamate di procedure sincrone e asincrone (2)

➤ Sincronizzazione client e server :

1. Applicazione appartenente ad un livello superiore del client e del server può iniziare lo scambio e controllare che tutte le azioni siano eseguite
2. Un client può emettere RPC asincrone seguite da una RPC sincrone finale alla quale il server risponderà solo dopo il completamento di tutto il lavoro richiesto dalle RPC asincrone

Bibliografia: Stallings - Sistemi Operativi

Sistemi per l'Elaborazione delle Informazioni

Topologie (1)

- ✓ I siti di un sistema possono essere collegati fisicamente in diversi modi differendo per :

➤ Costo di installazione

➤ Costo di comunicazione

tempo necessario per inviare un messaggio da un sito all'altro

➤ Disponibilità

in caso di guasti occorre stabilire se le comunicazioni sono ancora garantite

- ✓ Le varie topologie sono rappresentate tramite un grafo i cui nodi corrispondono ai siti. Ogni arco tra due nodi rappresenta una connessione diretta fra i due siti

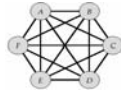
Bibliografia: Silberchatz, Galvin, Gagne - Sistemi Operativi, concetti ed esempi

Sistemi per l'Elaborazione delle Informazioni

Topologie (2)

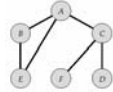
Una rete è :

- ✓ **Totalmente connessa**
se ogni sito è collegato a tutti gli altri siti



Costi di comunicazione più alto

- ✓ **Parzialmente connessa**
se esistono collegamenti diretti tra i siti ma non per tutti



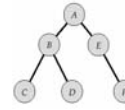
Costi di installazione minori

Bibliografia: Sibershatz, Galvin, Gagne – Sistemi Operativi, concetti ed esempi

Sistemi per l'Elaborazione delle Informazioni

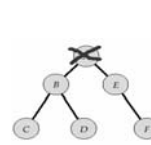
Topologie (3)

Rete a albero



- ✓ Bassi costi di installazione e comunicazione
- ✓ un guasto provoca un partizionamento

Un sistema si dice **partizionato** se è diviso in sottosistemi (partizioni) che non hanno connessioni tra di loro



Bibliografia: Sibershatz, Galvin, Gagne – Sistemi Operativi, concetti ed esempi

Sistemi per l'Elaborazione delle Informazioni

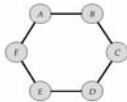
Topologie (4)

Rete a stella



- ✓ Un guasto provoca il partizionamento
- ✓ Il Guasto sito centrale provoca l'irraggiungibilità collasando la rete
- ✓ Basso costo di comunicazione: 2 collegamenti

Rete a anello



- ✓ 2 guasti provoca partizionamento
- ✓ C'è più disponibilità
- ✓ Alto costo per la comunicazione poiché attraversamento di più collegamenti

Bibliografia: Sibershatz, Galvin, Gagne – Sistemi Operativi, concetti ed esempi

Sistemi per l'Elaborazione delle Informazioni

Tipi di reti (1)

➤ Reti locali (local-area-network)

unità di elaborazioni distribuite su una piccola area

- ✓ cablaggi con doppini telefonici o fibre ottiche
- ✓ diffuse le reti con bus a multiaccesso, ad anello o a stella
- ✓ velocità ≈ 10 megabit al secondo, o più alta (Gigabit Ethernet).
- ✓ broadcast è più veloce ed economico
- ✓ i nodi sono di solito workstation e /o personal computer e pochi mainframe.

Bibliografia: Sibershatz, Galvin, Gagne – Sistemi Operativi, concetti ed esempi

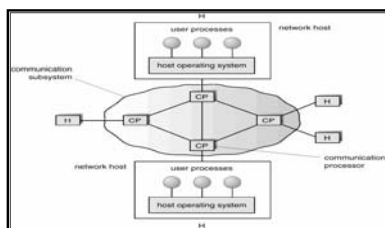
Sistemi per l'Elaborazione delle Informazioni

Tipi di reti (2)

➤ Reti Geografiche (wide-area-network)

distribuite su vaste aree geografiche

- ✓ Arpanet è stata la prima rete sperimentale, estesa sino ad diventare una rete mondiale di reti, l'Internet



Bibliografia: Sibershatz, Galvin, Gagne – Sistemi Operativi, concetti ed esempi

Sistemi per l'Elaborazione delle Informazioni

Tipi di reti (3)

➤ Le **unità di elaborazione della comunicazione** sono responsabili della definizione dell'interfaccia attraverso i quale i siti comunicano con la rete

➤ Utilizzo di **modem** per convertire segnali digitali in segnali analogici per WAN (ma anche per le LAN) che utilizzano normali linee telefoniche

➤ Utilizzo di **router** per determinare un instradamento dinamico, per aumentare l'efficienza o statico per ridurre i rischi ed aumentare la sicurezza

Bibliografia: Sibershatz, Galvin, Gagne – Sistemi Operativi, concetti ed esempi

Sistemi per l'Elaborazione delle Informazioni

A.A. 2004/2005

Comunicazione

Il progetto di una rete di comunicazione deve considerare quattro problemi fondamentali:

- Naming (risoluzione dei nomi)
- Strategie di routing (instradamento)
- Strategie riguardanti i pacchetti
- Strategie di connessione
- Contesa



Bibliografia: Silberschatz, Galvin, Gagne – Sistemi Operativi, concetti ed esempi Sistemi per l'Elaborazione delle Informazioni

A.A. 2004/2005

Comunicazione

Naming (risoluzione dei nomi) (1)

Affinchè due processi in esecuzione su due differenti macchine possano scambiarsi delle informazioni devono poter far riferimento l'uno all'altro
I sistemi in rete non condividono memoria quindi non hanno informazioni sui rispettivi processi in remoto

Risoluzione: ogni processo è identificato dalla coppia
<nome del calcolatore, identificatore>

Bibliografia: Silberschatz, Galvin, Gagne – Sistemi Operativi, concetti ed esempi Sistemi per l'Elaborazione delle Informazioni

A.A. 2004/2005


Comunicazione

Naming (risoluzione dei nomi) (2)

Nome del calcolatore: identificatore alfanumerico (anziché un numero) interno alla rete, per renderne semplice l'utilizzo da parte degli utenti.



Identificatore: PID o altro numero unico all'interno del calcolatore.



Bibliografia: Silberschatz, Galvin, Gagne – Sistemi Operativi, concetti ed esempi Sistemi per l'Elaborazione delle Informazioni

A.A. 2004/2005

Comunicazione

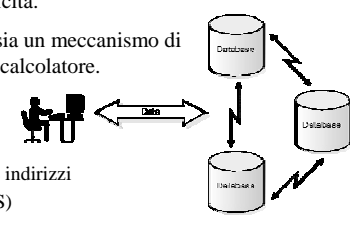
Naming (risoluzione dei nomi) (3)

All'interno della macchina il nome deve essere un numero per motivi di velocità e semplicità.

Per questo occorre che ci sia un meccanismo di **risoluzione** del nome del calcolatore.

Due possibilità:

- file di database con nomi e indirizzi
- domain name system (DNS)



Bibliografia: Silberschatz, Galvin, Gagne – Sistemi Operativi, concetti ed esempi Sistemi per l'Elaborazione delle Informazioni

A.A. 2004/2005

Comunicazione

Naming (risoluzione dei nomi) (4)

Il DNS specifica la struttura di nominazione dei calcolatori e la risoluzione dei nomi in indirizzi.

Basato su un database distribuito implementa il sistema di calcolo per la conversione da nome ad IP e da IP a nome.

La conversione avviene attraverso un sistema gerarchico, dove ogni nome è identificato da etichette separate da punti:

bob.cs.brown.edu

rappresenta un host sulla rete della Brown University.

Bibliografia: Silberschatz, Galvin, Gagne – Sistemi Operativi, concetti ed esempi Sistemi per l'Elaborazione delle Informazioni

A.A. 2004/2005

Comunicazione


Naming (risoluzione dei nomi) (5)

Partendo dalla fine della stringa, il sistema distribuito riconosce:

- brown.edu → dominio situato sul server dei nomi edu
- cs.brown.edu → identificato dal name server di questo livello
- bob.cs.brown.edu → identificato dal name server di cs, restituisce l'indirizzo IP della macchina (es. 128.148.31.100).

Prima dell'utilizzo del DNS?

Sistema SRI-NIC: file comune a tutte le macchine, da prelevare periodicamente.



Bibliografia: Silberschatz, Galvin, Gagne – Sistemi Operativi, concetti ed esempi Sistemi per l'Elaborazione delle Informazioni

A.A. 2004/2005

Comunicazione

Naming (risoluzione dei nomi) (5)

Gerarchia dei nomi

Autorità centrale

com edu org ... it

brown virgilio ... unisa

cs dia

bob

Bibliografia: Silbershatz, Galvin, Gagne – Sistemi Operativi, concetti ed esempi **Sistemi per l'Elaborazione delle Informazioni**

A.A. 2004/2005

Comunicazione

Strategie di routing (instradamento) (1)

L'instradamento di un pacchetto è il modo in cui avviene la trasmissione di un messaggio tra processi appartenenti a siti diversi.

Tre gli schemi di routing più diffusi:

- instradamento fisso
- instradamento virtuale
- instradamento dinamico

Bibliografia: Silbershatz, Galvin, Gagne – Sistemi Operativi, concetti ed esempi **Sistemi per l'Elaborazione delle Informazioni**

A.A. 2004/2005

Comunicazione

Strategie di routing (instradamento) (2)

L'**instradamento fisso** è molto semplice, ed è impostato di solito sul percorso (fisso) più breve tra A e B.

Pro: la scelta del percorso più breve riduce al minimo i costi di comunicazione.

Contro: in caso di guasti è difficile che i pacchetti arrivino a destinazione.

Bibliografia: Silbershatz, Galvin, Gagne – Sistemi Operativi, concetti ed esempi **Sistemi per l'Elaborazione delle Informazioni**

A.A. 2004/2005

Comunicazione

Strategie di routing (instradamento) (3)

L'**instradamento virtuale** è basato sullo stesso criterio di quello fisso, ma a differenza del precedente il percorso cambia ad ogni sessione.

Possiamo considerare una sessione come la durata dello scambio di messaggi (sessione di FTP, sessione di lavoro, ecc.)

Bibliografia: Silbershatz, Galvin, Gagne – Sistemi Operativi, concetti ed esempi **Sistemi per l'Elaborazione delle Informazioni**

A.A. 2004/2005

Comunicazione

Strategie di routing (instradamento) (4)

Nell'**instradamento dinamico** il percorso tra A e B è scelto al momento dell'invio del messaggio. A percorsi distinti sono assegnati messaggi distinti (quindi i messaggi possono non arrivare in ordine di invio).

La tecnica del routing dinamico è molto complicata da applicare, poiché ad ogni invio di solito si controlla la congestione di quel particolare canale.

Nelle **tavole di routing** di ciascuna macchina sono scritti i percorsi che si possono seguire per recapitare un messaggio (le tavole sono aggiornate ogni volta che la rete effettua un cambiamento).

Bibliografia: Silbershatz, Galvin, Gagne – Sistemi Operativi, concetti ed esempi **Sistemi per l'Elaborazione delle Informazioni**

A.A. 2004/2005

Comunicazione

Strategie riguardanti i pacchetti

Ogni messaggio può differire da un altro, per questo è preferibile suddividerli in *pacchetti* della stessa dimensione.

I *datagram* sono pacchetti scambiati tra due punti di una rete utilizzando un protocollo inaffidabile e privo di connessione (UDP).

I *segmenti* sono invece i pacchetti scambiati attraverso l'utilizzo di un protocollo di trasporto affidabile e orientato alla connessione (TCP).

Bibliografia: Silbershatz, Galvin, Gagne – Sistemi Operativi, concetti ed esempi **Sistemi per l'Elaborazione delle Informazioni**

A.A. 2004/2005

Comunicazione

Strategie di connessione (1)

Una volta stabilita la strategia per l'invio dei messaggi, si deve tener conto del tipo di schema di connessione che si intende applicare.

Sono tre gli schemi più diffusi:

- Commutazione di circuito
- Commutazione di messaggio
- Commutazione di pacchetto




Bibliografia: Silbershatz, Galvin, Gagne – Sistemi Operativi, concetti ed esempi Sistemi per l'Elaborazione delle Informazioni

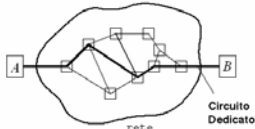
A.A. 2004/2005

Comunicazione

Strategie di connessione (2)

Commutazione di circuito

È il sistema utilizzato nelle comunicazioni telefoniche: non appena si stabilisce la comunicazione tra *A* e *B*, il circuito che si forma non può essere violato, anche a costo di bloccare temporaneamente quel canale di comunicazione.



Bibliografia: Silbershatz, Galvin, Gagne – Sistemi Operativi, concetti ed esempi Sistemi per l'Elaborazione delle Informazioni

A.A. 2004/2005

Comunicazione

Strategie di connessione (3)

Commutazione di messaggio

Se due processi vogliono comunicare, la comunicazione dura il tempo necessario della trasmissione del messaggio.

I collegamenti fisici sono temporanei (tempo di trasmissione) e dinamici.

Ogni messaggio contiene l'indirizzo d'origine e di destinazione, oltre al blocco del messaggio, e codici per la correzione degli errori, per far sì che il messaggio possa essere ricevuto integro.



Bibliografia: Silbershatz, Galvin, Gagne – Sistemi Operativi, concetti ed esempi Sistemi per l'Elaborazione delle Informazioni

A.A. 2004/2005

Comunicazione

Strategie di connessione (4)

Commutazione di pacchetto

Ogni messaggio può dividersi in più pacchetti, ognuno dei quali contiene informazioni di sistema (mittente, destinazione...).

A destinazione i pacchetti saranno ricomposti nei messaggi originari.



Bibliografia: Silbershatz, Galvin, Gagne – Sistemi Operativi, concetti ed esempi Sistemi per l'Elaborazione delle Informazioni

A.A. 2004/2005


Comunicazione

Contesa (1)

Secondo la tipologia della rete più di due host possono essere connessi tra loro condividendo la stessa linea. Questo può comportare collisioni tra i messaggi, quindi i mittenti devono essere informati per poter ritrasmettere i pacchetti andati perduti.

Soluzione:

- CSMA/CD
- Passaggio di contrassegno
- Intervalli di messaggi



Bibliografia: Silbershatz, Galvin, Gagne – Sistemi Operativi, concetti ed esempi Sistemi per l'Elaborazione delle Informazioni

A.A. 2004/2005

Comunicazione

Contesa (2)

CSMA/CD

Prima di trasmettere un messaggio un sito deve verificare che la linea sia libera. La tecnica è chiamata *Carrier Sense with Multiple Access – CSMA*.

Se, dopo aver trovato la linea libera, inizia la trasmissione ma, in contemporanea, altri siti trasmettono, si ha un rilevamento di collisione (*collision detection – CD*), con istantanea interruzione della trasmissione.

È impiegato nel protocollo 802.3 (*Ethernet*).

Bibliografia: Silbershatz, Galvin, Gagne – Sistemi Operativi, concetti ed esempi Sistemi per l'Elaborazione delle Informazioni

A.A. 2004/2005

Comunicazione

Contesa (3)

Passaggio di contrassegno (*token ring*)

Utilizzata nelle reti ad anello, un pacchetto “contrassegno” circola in continuazione tra gli host, ai quali è concessa la trasmissione solo dopo la ricezione del token (rimosso in trasmissione e riattivato alla fine).

Se il token viene perso c'è un sito predisposto all'invio di uno nuovo.

Utilizzato nei sistemi IBM e HP/Apollo.

Bibliografia: Silberhatz, Galvin, Gagne – Sistemi Operativi, concetti ed esempi Sistemi per l'Elaborazione delle Informazioni

A.A. 2004/2005

Comunicazione

Contesa (4)

Intervalli di messaggi

Nel sistema circola continuamente un certo numero di messaggi “vuoti” (*intervalli*) in cui un host può inserire il proprio messaggio (completo di mittente e destinatario). Alla ricezione l'intervallo viene svuotato per permettere agli altri siti di inviare informazioni.

Tutti gli host controllano il contenuto dell'intervallo pieno.

Utilizzato nel sistema *Cambridge Digital Communication Ring*.

Bibliografia: Silberhatz, Galvin, Gagne – Sistemi Operativi, concetti ed esempi Sistemi per l'Elaborazione delle Informazioni

A.A. 2004/2005

Protocolli di comunicazione (1)

La rete di comunicazione, secondo lo standard OSI della ISO, è divisa nei seguenti strati o livelli (*layers*):

- **Strato fisico** – responsabile del funzionamento meccanico ed elettrico della rete, si occupa della trasmissione fisica del flusso di bit.
- **Strato data-link** – responsabile della gestione di *frames* (o di parti di lunghezza fissa), comprende l'individuazione e la correzione degli errori verificati nello strato fisico.
- **Strato di rete** – gestore di collegamenti e dell'instradamento di pacchetti nella rete di comunicazione, è responsabile del controllo degli indirizzi dei pacchetti in uscita.

Bibliografia: Silberhatz, Galvin, Gagne – Sistemi Operativi, concetti ed esempi Sistemi per l'Elaborazione delle Informazioni

A.A. 2004/2005

Protocolli di comunicazione (2)

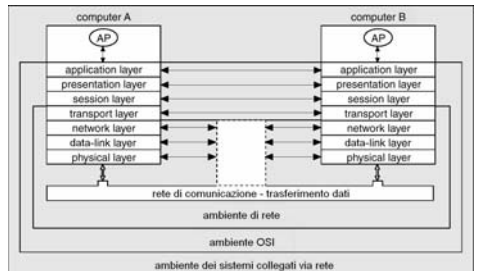
- **Strato di trasporto** – gestore del trasferimento dei messaggi tra host (suddividendoli in pacchetti) controlla il flusso e la generazione degli indirizzi fisici. 
- **Strato sessione** – responsabile della creazione delle sessioni e delle comunicazioni tra processi, gestisce il trasferimento di files e la posta elettronica.
- **Strato di presentazione** – gestisce la trasformazione dei formati che si possono presentare tra diversi siti della rete
- **Strato di applicazione** – è responsabile della comunicazione con l'utente. Tratta l'FTP, i protocolli di gestione di sessioni di lavoro remote, posta elettronica e schemi per basi di dati distribuite.

Bibliografia: Silberhatz, Galvin, Gagne – Sistemi Operativi, concetti ed esempi Sistemi per l'Elaborazione delle Informazioni

A.A. 2004/2005

Protocolli di comunicazione (3)

Immaginiamo gli strati del protocollo come una pila, ognuno dei quali comunica logicamente con il suo corrispondente, associato agli altri sistemi:



Bibliografia: Silberhatz, Galvin, Gagne – Sistemi Operativi, concetti ed esempi Sistemi per l'Elaborazione delle Informazioni

A.A. 2004/2005

Protocolli di comunicazione (4)

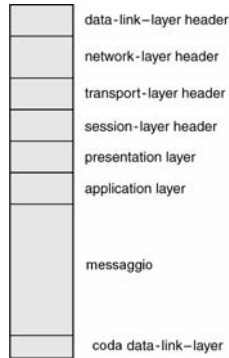
In realtà ogni messaggio (che di solito parte dal livello applicazione del mittente per arrivare al livello applicazione del destinatario)  passa:

- dallo strato più alto al livello fisico in invio,
- dallo strato fisico al livello applicazione in arrivo.

Bibliografia: Silberhatz, Galvin, Gagne – Sistemi Operativi, concetti ed esempi Sistemi per l'Elaborazione delle Informazioni

Protocolli di comunicazione (5)

Il messaggio di rete OSI contiene quindi un *header* per ogni livello, a partire dal data-link fino al livello applicazione (in ordine inverso), escluso il livello fisico (che vede direttamente il flusso di bit), e una coda che chiude il messaggio, associata allo strato data-link.



Architettura del protocollo TCP/IP (1)

Per il TCP/IP non esiste un modello ufficiale, ma esiste comunque una stratificazione in cinque livelli:



- Livello fisico
- Livello di accesso alla rete
- Livello di internetworking (IP level)
- Livello di trasporto (TCP - *Transmission Control Protocol*)
- Livello applicazione

Architettura del protocollo TCP/IP (2)

I due livelli più bassi (fisico e di rete) sono implementati in modo simile allo standard OSI, così come il livello applicazione (che include sia presentazione che applicazione di OSI).

IP gestisce le funzioni d'instradamento su reti multiple ed è implementato su tutti i sistemi e router. Agisce come collegamento per spostare un blocco da un host ad un altro attraverso router.

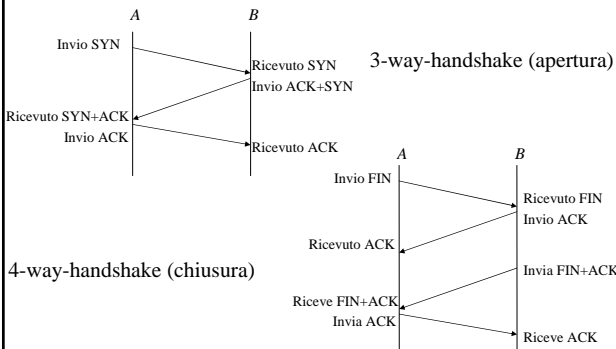
Architettura del protocollo TCP/IP (3)

TCP si occupa del trasporto host-to-host ed è un protocollo di trasporto **affidabile ed orientato alla connessione**. Tiene traccia dei blocchi dei dati per assicurarsi l'avvenuta consegna.

Per garantire questo servizio fa uso di una tantissimi controlli, e in particolare stabilisce la connessione tramite due operazioni di *handhaking*:

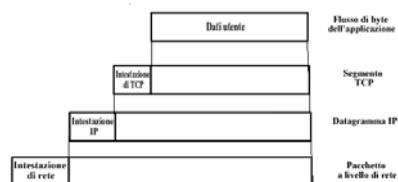
- 3-way-handshake (apertura)
- 4-way-handshake (chiusura)

Architettura del protocollo TCP/IP (4)



Architettura del protocollo TCP/IP (5)

Insieme ai dati sono aggiunte delle informazioni nelle intestazioni (*header*) di tutti gli altri livelli, utilizzate per il controllo dell'operazione da fare:



Architettura del protocollo TCP/IP (6)

Elementi inclusi nell'intestazione del segmento TCP:

- > Porta di destinazione
per sapere a chi inoltrare i dati
- > Numero di sequenza
i segmenti vengono numerati per evitare eventuali errori di ricezione
- > Checksum
inclusione di un codice di 16 bit che è funzione del contenuto della parte rimanente, per evitare eventuali errori di trasmissione

In seguito il TCP consegna ogni segmento all'IP, con l'istruzione di trasmetterlo al destinatario (con la necessità di inserire una nuova intestazione per formare il datagram IP)

Architettura del protocollo TCP/IP (7)

Infine ogni datagram IP viene presentato al livello di accesso alla rete per essere trasmesso alla prima sottorete. Anche questo livello appende la sua intestazione, contenente:

- > indirizzo della sottorete di destinazione
- > altre funzionalità richieste
(funzionalità della sottorete come per esempio le priorità)

Quando i dati arrivano alla destinazione si effettua il processo inverso, ed ad ogni livello l'intestazione viene rimossa e la parte restante passata al livello successivo, sino alla consegna dei dati originali

Comparazione: OSI e TCP/IP

Vediamo la corrispondenza tra i livelli del modello ISO/OSI e TCP/IP

Le "sessioni" sono create nel livello TCP o Application a seconda delle necessità

OSI	TCP/IP
Application	Application
Presentation	Application
Session	Application
Transport	Transport (host-to-host)
Network	Internet
Data Link	Network Access
Physical	Physical

Il livello applicazione abbraccia per intero i due livelli più alti dell'OSI

Il livello fisico è l'unico che coincide integralmente

UDP (User Datagram Protocol) (1)

UDP è un servizio che fornisce un servizio inaffidabile e senza connessione per procedure a livello di applicazione. Rispetto al meccanismo di TCP è molto più semplice da implementare ed è molto più veloce.

Svantaggi:

- > Non garantisce la consegna
- > Non rispetta la sequenza di frammentazione
- > Non si accorge dell'invio di duplicati. L'UDP abilita una procedura a spedire messaggi ad altre procedure mediante un meccanismo di protocollo minimo

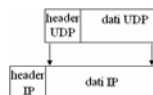
L'UDP fornisce essenzialmente all'IP la capacità di indirizzamento delle porte

UDP (User Datagram Protocol) (2)

UDP identifica le applicazioni mediante l'utilizzo di numeri di porta a 16 bit.

È utilizzato in applicazioni che mirano alla velocità del servizio (come il DNS).

È semplicemente un'estensione di IP, con l'aggiunta delle porte di invio e destinazione.



Quando usarlo: UDP è consigliato per applicazioni sviluppate in reti locali, con scambio di messaggi di piccole dimensioni (senza bisogno di segmentare in pacchetti).

TCP vs UDP (headers)


Intestazione TCP

Porta sorgente	Porta destinazione
numero di sequenza	
numero di acknowledgement	
useless	flag
checksum	finestra
puntatore urgente	
opzioni e riempimenti	

Intestazione UDP

Porta sorgente	Porta destinazione
lunghezza segmento	checksum

A.A. 2004/2005




Robustezza

Ogni sistema distribuito può soffrire di vari tipi di guasti fisici. La robustezza della rete indica quanto il sistema sia in grado di continuare la trasmissione dei dati tramite tre operazioni:

- rilevamento dei guasti
- riconfigurazione del sistema
- ripristino del collegamento

Bibliografia: Silbershatz, Galvin, Gagne – Sistemi Operativi, concetti ed esempi Sistemi per l'Elaborazione delle Informazioni

A.A. 2004/2005




Robustezza

Rilevamento dei guasti (1)

È difficile distinguere tra guasti relativi al collegamento, all'host o alla perdita del messaggio.


Per rilevare un guasto i siti utilizzano una procedura di negoziazione detta *handshake* (come TCP), caratterizzata dall'invio di messaggi del tipo:

- *I-am-up*
- *Are-you-up?*



Bibliografia: Silbershatz, Galvin, Gagne – Sistemi Operativi, concetti ed esempi Sistemi per l'Elaborazione delle Informazioni


A.A. 2004/2005



Robustezza

Rilevamento dei guasti (2)


Supponiamo un collegamento fisico tra *A* e *B*.
A intervalli fissi entrambi gli host si inviano un *I-am-up*.
A può sopporre la presenza di un guasto se non riceve questo messaggio da *B* in un certo intervallo temporale.

Due possibilità per *A*: attendere ancora o inviare un *Are-you-up?* 

Se *A* non riceve risposta il guasto è verificato di sicuro.

Bibliografia: Silbershatz, Galvin, Gagne – Sistemi Operativi, concetti ed esempi Sistemi per l'Elaborazione delle Informazioni

A.A. 2004/2005



Robustezza


Rilevamento dei guasti (3)

A può cercare di intuire il guasto inviando *Are-you-up?* Per un altro percorso. Se arriva la risposta di *B* entro un intervallo di tempo allora la comunicazione può continuare, altrimenti *A* può concludere che si è verificata una delle seguenti situazioni (non determinabile):

- *B* è fuori servizio;
- il collegamento è fuori servizio;
- il percorso alternativo è fuori servizio;
- il messaggio è andato perduto.

Bibliografia: Silbershatz, Galvin, Gagne – Sistemi Operativi, concetti ed esempi Sistemi per l'Elaborazione delle Informazioni

A.A. 2004/2005



Robustezza


Riconfigurazione del sistema

Supponendo che *A* abbia scoperto che si sia verificato un guasto. Serve una procedura che permetta al sistema di continuare la trasmissione:

- se si è verificato il guasto sul collegamento bisogna riaggiornare le tavole di routing di tutti i siti in rete
- se si è verificato il guasto sul sito bisogna informare gli host che non possono usufruire dei servizi del sito guasto (se il sito era coordinatore bisogna eleggere il nuovo).

Bibliografia: Silbershatz, Galvin, Gagne – Sistemi Operativi, concetti ed esempi Sistemi per l'Elaborazione delle Informazioni

A.A. 2004/2005



Robustezza

Ripristino del collegamento


Quando un collegamento o un sito guasto viene riparato deve essere reintegrato nel collegamento in modo semplice e lineare:

- riparato il collegamento tra *A* e *B* bisogna informare gli host della riparazione (ripetendo l'*handshaking*)
- riparato il sito *B*, questo deve informare gli altri del suo ritorno e aggiornare le tavole di routing relative alla nuova configurazione

Bibliografia: Silbershatz, Galvin, Gagne – Sistemi Operativi, concetti ed esempi Sistemi per l'Elaborazione delle Informazioni

A.A. 2004/2005

Problemi di progettazione (1)



Il principale scopo dei progettisti è rendere **trasparente** agli utenti la molteplicità delle unità di elaborazione.


L'ambiente dovrebbe apparire come un sistema centralizzato in cui:

- poter aprire sessioni da un qualsiasi calcolatore
- continuare sessioni aperte da qualsiasi stazione
- utilizzare gli stessi file e dividerli con altri utenti

Bibliografia: Silbershatz, Galvin, Gagne – Sistemi Operativi, concetti ed esempi Sistemi per l'Elaborazione delle Informazioni

A.A. 2004/2005

Problemi di progettazione (2)



Il sistema deve essere **tollerante ai guasti**, dove per guasti (tollerabili) si intendono rotture di dispositivi di memoria, errori di comunicazione, guasti di macchina (*fail-stop*).

La **scalabilità** del sistema è il punto cardine nella progettazione (concetto non utilizzabile in senso assoluto).

Non esistono sistemi scalabili, esistono sistemi più scalabili di altri, in quanto non può esistere un sistema in cui non si crei mai intasamento, o che possa effettuare calcoli alla stessa velocità con due o cento processi.

Bibliografia: Silbershatz, Galvin, Gagne – Sistemi Operativi, concetti ed esempi Sistemi per l'Elaborazione delle Informazioni

A.A. 2004/2005

Un esempio di comunicazione in rete (1)

- Trasmissione di un pacchetto tra host di una rete Ethernet.
- Ogni host ha un indirizzo IP unico e un corrispondente indirizzo Ethernet (MAC – *medium access control address*).
- La comunicazione richiede entrambi gli indirizzi.
- Il DNS può essere utilizzato per acquisire gli indirizzi IP.
- L'*Address Resolution Protocol* (ARP) è utilizzato per mappare il MAC address sull'indirizzo IP.
- Se gli host sono sulla stessa rete l'ARP può essere usato, altrimenti il sito che invia il pacchetto dovrà inviare un pacchetto al router che instrada il messaggio sulla rete di destinazione.

Bibliografia: Silbershatz, Galvin, Gagne – Sistemi Operativi, concetti ed esempi Sistemi per l'Elaborazione delle Informazioni

A.A. 2004/2005

Un esempio di comunicazione in rete (2)

bytes		
7	intestazione - inizio del pacchetto	configurazione byte: 10101010
1	delimitatore di segmento	configurazione di bit 10101011
2 o 6	indirizzo destinazione	indirizzo ethernet o broadcast
2 o 6	indirizzo sorgente	indirizzo ethernet
2	lunghezza della sezione dati	lunghezza in byte
0-1500	dati	message data
0-46	byte di riempimento	la lunghezza del messaggio deve essere > 63 byte
4	checksum	per la rilevazione degli errori

Bibliografia: Silbershatz, Galvin, Gagne – Sistemi Operativi, concetti ed esempi Sistemi per l'Elaborazione delle Informazioni