

sistemi operativi anno 2021-2022 registrazioni: 2° semestre

2022-03-03

slide 01-arch-hw.pdf

Granotteria di oggi: "Ponte Giappone di Monet" qual'è il messaggio? È sempre impressionista. Ma il messaggio è più nel ponte oggi. Bisogna costruire ponti. Muoi i ponti e altre cose distinguono il vivere. Città, i punti uniscono. Non a caso i popoli vivono opposti chiamati "frontifici" che creano ponti. Ma. Pensiamo a Sistemi Operativi. Avete dubbi o domande?

Riaccanto delle puntate precedenti:

Stiamo iniziando a guardare come costruire Sistemi Operativi; mentre prima abbiamo visto come usare i loro.

E diciamo guardando di sapere alcuni particolari relativi all' architettura dell' hardware. Nella scena che abbiamo visto un po' di lucidi spioni, abbiamo visto che il sistema operativo è quello che crea una aborazione sull' hardware.

Quindi se volete costruire un Sistema Operativo dovete sapere come funziona l' hardware, quali è il linguaggio parlato dall' hardware. Abbiamo visto dall' inizio di questa lezione di lucidi cosa sono gli interrupt qui il Reader gli interrupi sono così importanti? e che diventano: sicuramente il modo per funzionare i sistemi operativi. Andiamo poi a vedere quelle che sono le risorse che il sistema operativo deve gestire.

Era comunque arrivato qua (slide 03 01-arch-hw.pdf) scrivendo bene. Una risorsa molto importante, fondamentale è la RAM (Random Access Memory) per come sono costituiti i sistemi

Oggi la RAM è la memoria veloce, direttamente accessibile al processore volatile. Non sempre i sistemi erano fatti così, il primo sistema che ho usato nella mia vita aveva una memoria ai nuclei di ferrite, e quindi quello che era la memoria di lavoro era non volatile, era permanente. Mentre invece nei sistemi comuni oggi si tende a usare la RAM. La RAM assieme ai registri è l'unico operario di memorizzazione accedita direttamente dal processore. Per accedere a memoria secondaria e terziaria ed altre risorse, non lo si fa direttamente parlando con il bus, ma bisogna tramite il bus dare dei comandi a dei controller che accedono al device. La RAM nei sistemi moderni è chiamata RAM statica, viene costruita con praticamente dei condensatori che si caricano, deve essere rinfrescata il contenuto perché tende a scempare con il tempo, ma tutta questa attività è invisibile al Kernel perché viene fatta direttamente dai moduli della RAM stessa. Quindi per quanto riguarda l'uso della RAM si accede direttamente al bus mettendo l'indirizzo a cui si vuole leggere e scrivere e si da il comando LOAD / STORE. Nei sistemi moderni, però, l'indirizzamento della RAM non viene attuato in maniera diretta ma tramite un'unità fisica, un componente fisico chiamato Memory Management Unit (MMU). Mi raccomando per le abbreviazioni e i concetti similari: Esiste una cosa chiamata Memory Mapper, che è una componente del sistema operativo e del software. Esiste una cosa chiamata Memory Management Unit che è una componente hardware ed è il componente che interfaccia la memoria.

Se avete il Memory Manager è il componente del sistema operativo che pilota, programma, configura la MMU. ok.

Esistono ancora nei sistemi delle ROM (Read Only Memory) pensate a spron o cmos ram, alimentate, cioè memoria normalmente utilizzate in sola lettura. Tipicamente queste memorie vengono utilizzate per funzionalità molto basilari, per esempio per fare il boot della macchina. Lo avete un po' visto anche quando avete visto un po' di architettura di romps (microumps).

Esistono alcuni dispositivi che sono MEMORY MAPPED, che vi riconosce ad accedere come se fossero memoria, ad esempio il video grafico del Personal Computer. Ad alcune indirizzi di bus corrispondono delle aree che non sono vere aree di memoria, ma dove il sistema può scrivere dei dati e questi dati vengono letti in tempo reale. È una memoria a doppio accesso: il bus scrive da un lato e dall'altro vengono letti da un componente, la scheda grafica, che prende questi valori scritti come valori normalmente come luminosità RGB (Red Green Blue, i tre colori) per accendere i vari pixel sullo schermo. Lo schermo grafico viene accedito in questo modo la gestione è semplice e lineare perché in questa modalità una su collocare l'indirizzo di quel pixel, scrivere un valore e il valore diventa una luminosità di un pixel. Ma dall'altra parte necessita di sincronizzazione di accesso. Essendoci due componenti che contemporaneamente accedono a questa memoria a doppio accesso, quello

che succede è che ci vogliono meccanismi di sincronizzazione
talvolta fatti in maniera hardware perché ad esempio
il video non faccia FLICKERING (Dizionario Oxford:
Sostantivo, lo spavallio dell'immagine salto schermo
di una televisione o di un computer). Quindi questo c'è
una doppia memoria, viene scritto il nuovo frame noi
ai dà l'impronta per fare in modo che il nuovo
frame diventi quello visibile dall'utente. (pagg 25)
DISCHI! Nonostante l'avvento, momentaneamente, della presenza di
dischi: allo stato Solid State Drive (SSD) si usano
tuttissime ancora i dischi rotazionali; si chiamano conservazionali;
quelli: che abbiamo visto alla prima lezione di questo
semestre direttamente aperti. I dischi sono dispositivi di memorizzazione
l'informazione in forma magnetica e quindi mantengono l'informazione
in maniera non volatile, significa che anche se togliete l'alimentazione
potete ritrovare i dati scritti. L'**accesso** è diretto (random i.e. non
sequenziale). Per capire questa affermazione bisogna vedere che
esistono anche dei dispositivi magnetici ad accesso sequenziale,
che sono i nastri, i tape (esiste ad accesso sequenziale).
L'indirizzamento su disco è dato in **termini di 3 parametri**:
cilindri, settori e testine. Per cambiare cilindri, biscece lo
affidate ad architettura, quindi non sto a spiegare. Tutto,
c'è ilsettore delle testine e quindi se noi non muovete il
settore delle testine il disco gira e quindi tutto ciò che è
memorizzato con quella posizione della testine è accessibile, nonoi
aspettando che durante una rotazione transitati il settore sotto lo

testina, usci decidendo di uscire una delle testine fra quelle disponibili. Se invece uno vuole apportare il prezzina delle testine, le quindi cambierà cilindro si chiama cilindro perché ovviamente se tenete la testina ferme il disco gira tutti i dati saranno a una certa distanza dal punto di rotazione, quindi se lo pensate geometricamente è un cilindro. Per cambiare cilindro ci vuole un tempo meccanico di spostamento della testina. Per combinare sette ci vuole un tempo meccanico per aspettare che quel dato che vogliamo leggere e scrivere passi sotto la testina. Per cambiare testina non ci vuole tempo meccanico basta tempo elettronico. Per intenderci per cambiare cilindri occorrono dei tempi dell'ordine di millisecondi e dipende anche da quanti cilindri dorete voltare, per cambiare settore il tempo si può anche calcolare, perché quando comprate un disco il disco fa i parametri di lavoro a il numero di giri al secondo. Se avete un disco a basso costo oppure un disco che fa 5400 giri al minuto RPM (revolutions per minute) e se prendete un disco molto costoso ne fa 15000, non è che possiate avere un disco che fa 2000000 giri al minuto, decollare il computer: D'ao, non ce la si fa fisicamente a costituire. Quindi se pensate 5400 giri :: facciamo 600 per fare i conti passi se poi fanno 6000 giri al minuto al secondo sono 100, quindi vuol dire 10 millisecondi, quindi anche quelli non è un tempo banale. ok? Dal punto di vista del sistema operativo questo cosa significa? Perché non è indolare avere questa gestione. Significa che le operazioni di SEEK devono essere limitate e bisogna fare in modo

(page 26)

che siano brevi. Quindi quello che succede è che, visto che i vari processi chiederanno in concorrenza di voler leggere e scrivere dati su disco occorrerà avere anche in questo caso dei meccanismi di Scheduling di ordinamento delle richieste che ottimizzino il percorso della testina su disco. ok? È altra cosa, che incide più sull'implementazione del file-system rispetto alla gestione fisica del disco. È quello di fare in modo che dati che vengono accediti spesso, in sequenza o vicini nel tempo siano vicini anche su disco. Se noi utilizzi dei file-system, e lo vedremo, poco efficienti, costituiti male per esempio FAT (File Allocation Table), a forza di usare quel file system spargerà i "pezzi" dei file "in posti casuali" del disco. ok? E quindi farà in modo che il sistema man mano perdga tempo di ricerca di posizioni. ok? Infatti sono quei file system per i quali è prevista la fragmentation. Non so se avete mai fatto defrag. ma potrete scoprire altri file system che non avevano questo tipo di problema. Mi vengono in mente 2 cose, che mi faccio vedere per rompere il ritmo.

Se google cerco "cifrare hard disk espazio" clicca su "Trappola del Cyberspazio - Roberto di Cosimo" link: <https://www.dicosmo.org/cybersec/trapools.htm>

Roberto di Cosmo, menziona il nome. è italiano, non vive e lavora in Francia, insegna diritti a Paris Diderot, (legge il capitolo intitolato "scandal: a cassetto" e leggiro dei cencelli: ") e racconta: "Ebbene durante uno dei miei viaggi mi sono ritrovato a fianco di un gentilissimo signore, giovane e dinamico

funtionario d'impresa, che si apprestava ad eseguire sulla sua macchina il famigerato programma DeBragg. Questo programma metteva una delle matrice riempita di piccoli quadrati di toni colori che si muoveva in tutte le direzioni mentre il disco lavorava intensamente e rumorosamente.

Non ho potuto resistere alla tentazione (...): dopo essermi complimentato per il suo bel portatile, gli ho chiesto, fingendo la più grande ignoranza, [di cosa fossé quel bellissimo programma che io non avevo sul mio portatile. Con un air di superiorità mista a compassione (...)] mi ha risposto che si trattava di uno strumento essenziale che bisogna lanciare di fatto in tante per "fare andare la macchina più veloce". Ha proseguito ripetendomi a memoria gli argomenti che si trovano nei manuali Windows: più si utilizza il disco, più questo si "frammumenta", e più il disco è frammentato più la macchina è lenta.

A questo punto ho tirato fuori il mio computer portatile, che non utilizzo Windows, ma GNU/Linux (...), e gli ho detto con un'aria un po' stupita, che tutto quello che mi aveva detto mi comprendeva enormemente: sul mio portatile il disco è molto poco frammentato e più si utilizza meno si frammenta. [Questo disco è come una gigantesca armadio a cassetti: ogni cassetto ha la stessa capacità e ciascun disco contiene alcuni milioni di cassetti. Cassette di quelli di oggi: 7. Se i dati che vi interessa sono sistemati in cassetti contigui, si può accedere più velocemente (...].

In magazziniamo un ministero che conserva i suoi dossier in un enorme armadio con milioni di cassetti [..].

Se andate nel segretario [lo metto al maschile perché lo trovo più coerente...] con due candidati dalle abitudini molto diverse uno che guarda tutto in dossier si limita a restare in cassetti e guarda troppo uno nuovo lo separa in piccoli fascicoli e li mette a cose nel primo cassetto vuoto che trova. Un altro che conserva sulla sua scrivania una lista di cassetti vuoti contigui e aggiorna la lista tutte le volte che la pratica viene chiusa. Questo lo trova una maniera molto simpatica di raccontare la frammentazione.

L'altra cosa che mi sembra far vedere (ed è solo troppo) (cerca su google "lest we remember :Cold Boot Attacks on Encrypted Keys" link: https://www.usenix.org/legacy/event/sec08/tech/full_papers/heldemann_heldemann.pdf)

<https://cipest.s3.amazonaws.com/up-content/uploads/2019/01/23195456/heldemann.pdf>

"Voi pensate che la RAM sia volatile, in realtà qui i hanno fatto vedere un attacco su chiavi crittografiche fatte sulle RAM in chi modo? Prendendo la RAM in funzione e facendo queste ottime velocissime, e poi neanche troppo veloci, tempi qualche secondo:

- Significato 'diciottembre' la macchina senza che il sistema nostra fare shutdown.

- Togliete la RAM e consigliatela
- E ancora possibile leggere quello che c'è scritto sulla RAM.

Interessante... ecco (page 6, figura 4 dell'articolo)

Quello è l'immagine nella RAM e vedete il contenuto dopo 5 secondi: • 30 secondi: • 60 secondi: • 5 minuti.
Quindi: compiuta, entro 5 secondi avete ancora tantissime informazioni utili per vedere (ovviamente era un immagine ma l'hanno fatta vedere su chiari crittografiche)..., in fondo diari crittografiche quando noi scrivete la password, è vero che nel interno viene salvata in maniera crittografata e così via, ma per fare l'input della password quella password per un po' passa in RAM, in chiaro." Comunque per nostra informazione non penso vi sia utile per fare Sistemi Operativi a prova di congelamento della RAM.

Quando si dice che la RAM è volatile è proprio perché è fatta con i condensatori, quindi i condensatori si scaricano pian piano, non c'è più il meccanismo di refresh perché non c'è più corrente, ma il contenuto può essere recuperato.

OK. (mag 27)

S.S.D. Ho aggiunto questo lucido perché anche SSD ha degli effetti collaterali sui sistemi operativi. Per copiare cosa dobbiamo fare con i sistemi operativi dobbiamo copiare qualcosa caratteristica alle quali dobbiamo stare attenti: di queste associazioni. Il numero di cicli di scrittura è limitato, altrò non limitato, quindi occorre per quanto possibile spargere uniformemente sulla gamma di indirizzi, cioè nello spazio memorizzabile de operazioni. Quindi è bene riciclare lo spazio utilizzatore in maniera da... Non importa come noi diciamo cercarla vicina, ma occorre fare in modo di riciclare e riutilizzare ora ciò che abbiamo utilizzato in tempo più temuto,

in maniera tale da spargere l'accesso su tutti gli indirizzi.

Sì leggono a blocchi Si scrivono a blocchi (memoria bloccata)

Se mechanismo di scrittura dell'SSD prevede che la scrittura non venga fatta per un blocco solo ma per un gruppo di blocchi ok? Queste cose sono normalmente allontanata mostrerai dai controller, ma meglio si vanno queste cose, per esempio è bene fare in modo che cose che vengono aggiornate contestualmente siano a indirizzi limitati. Per il numero di cicli di scrittura limitato vedremo che ci sono degli effetti collaterali anche non solo nella gestione fisica del dispositivo ma anche sul file system, per esempio un caso tipico d'uso è quello di writer di fare aggiornamenti non necessari, grosse cache aiutano, ma tra l'altro nei sistemi UNIX, per esempio, è possibile, quando si monta un file system, dichiarare se si vuole avere la data di ultimo accesso al file o no. In realtà la data di ultimo accesso ad un file, accesso non modifica, se togliete la data di modifica soltanto un solo di cose per esempio make non funziona più. Mentre invece lo dato di ultima lettura è normalmente utile, quindi di solito se avete dei dischi SSD si fa in modo che non venga aggiornata, c'è il parametro di mount che si chiama rottime, non access time, che fa in modo che non ci sia questa informazione aggiornata. Pensate "Quanti file leggete?" "Quanti file scrivete?" "Quante volte leggete un file?" pensate /etc/passwd: tutte le volte che c'è un autenticazione da qualche parte e si vuole vedere qual'è il nome corrispondente a un utente c'è un accesso a quel file.

Quanto più velocemente degrada un disco allo stato solido (SSD) ci chiedete questo aggiornamento? Perché ovviamente è dato di controllare continuamente e essere aggiornati ad ogni accesso, o quanti tenti di fare cache degli indirizzi usati per... (pag. 28)

Quindi possiamo pensare di avere una gerarchia di memoria, abbiamo tante memorie, abbiamo registri, RAM, dischi e anche unità offline.

Andrew Tanenbaum ha detto

"Never underestimate the bandwidth of a station wagon full of tapes hurtling down the highway"

e il prof lo ha tradotto così:

"Non sottovalutate mai il bandwidth di un veicolo necessario carico di moduli"

Google Traduttore ha tradotto così:

"Non sottovalutate mai la larghezza di banda di una stazione wagon piena di nastri che affrezza lungo l'autostrada."

In realtà il bandwidth può essere moltissimo, è difficile accederlo...

E cosa c'entra con questo? Di solito le memorie vengono chiamate memoria primaria, secondaria, terziaria. È prima ancora della memoria primaria ci sono i registri da memoria primaria e quella direttamente accessibile dal processore, quindi normalmente la RAM. La memoria secondaria è la memoria accessibile tramite un controller del processore quindi i dischi rigidi e i SSD. La memoria terziaria è memoria offline che può essere collegata in necessità, la memoria terziaria normalmente ha bisogno di un intervento umano o

simili per farlo.

bot: che cosa intende per ciclo di scrittura?

Per lo SDD, quando dovete aggiornare il dato che avete memorizzato sullo SDD dovete dare un comando di scrittura e quindi quello è un ciclo di scrittura, il ciclo di scrittura fisico.

Come dicono prima normalmente per farne il meno possibile ai tempi dei buffer, quindi se un dato un piccolo file viene accedito, modificato più volte in realtà non si fa alcuna scrittura fisica, se non in fondo se il file non viene anche cancellato. OK? Quindi l'idea è quante scritture fate, lo chiamo ciclo di scrittura e non scrittura perché è diversa la scrittura che appreso a voi quando avete un programma fate una scrittura e la scrittura fisica che viene fatta poi sicuramente sulla SDD. Come le chiavette USB, che in fondo sono fatte più o meno della stessa pasta, se mai provate a usarla pesantemente in lettura e scrittura e la staccate senza fare l'operazione di chiusura di umount su quello chiavetta ci può essere di tutto, perché tenderà a fare molte scritture possibili fisiche e molte cose quando l'andate a scollegare pensavate di avere fatto una tana ancora nel buffer.

Allora abbiamo una gerarchia di memoria (prof 29, figura)

Partendo dai registri guardiamo una riga o è una riga no. **registri, memoria principale** che è la RAM; **memoria secondaria Disco SDD; memoria temporaria**. Poi c'è una linea di demarcazione tra la memoria volatile e quella non volatile, ma oltre gli elementi dinamici nel senso ci sono gli elementi fissi che sono le **CACHE**. L'ultima parte l'ha un po' inventata io per

domandarmi che cosa c'è oggi d'equivalente delle memorie terzarie.

In fondo oggi come memorie terzarie quello che si usa più normalmente sono memorie allo stato solido removibili, mentre una volta c'erano nastri, CD rom, DVD rom che hanno prodotto sempre più utilizzo perché risultano più costosi, più scomodi di unità allo stato solido. Parliamo del concetto di cache importantsissime.

Che cos'è una CACHE?

Una cache è una porzione di memoria più veloce che viene usata per mantenere temporaneamente i dati più utili di una memoria più lenta. Facendo così statisticamente potrai capire frequentemente di poter ritrovare il dato senza andare ad accedere alla memoria più lenta, ma trovandolo già disponibile nella memoria più veloce. Ok? Questo è un concetto abitato da poche applicazioni: i livelli: per esempio nella CPU nei processori ci sono delle cache del contenuto della memoria, se guardate quando acquistate un processore o acquistate una macchina con dentro un processore, nelle caratteristiche tecniche del processore c'è scritta quanta cache c'è: ok? Questa cache che vedete

la in [\(figura n. 2.9\)](#) tra i registratori e la memoria centrale è costruita con le tecnologie dei registratori quindi molto veloce e questa cache è gestita direttamente dal processore quindi normalmente è trasparente ai sistemi operativi. Quindi: il sistema operativo / il programma applicativo non si accorge della cache se non dal punto di vista prestazionale, quando si accede alla memoria centrale automaticamente la cache fa il suo mestiere e tenta di mantenere i dati più recentemente usati. Tutti questi meccanismi si basano su un principio di ... viene spesso detto in informatica di località, se possibile. S'è quindi informatici dell'intera. Ci si aspetta che dati: limitati

in memoria, logicamente limitrophi: in memoria e dati recentemente utilizzati: vengono utilizzati nel prossimo futuro, che non è una regola generale. È possibile costruire dei programmi per fare in modo che la cache lavori peggio possibile, però statisticamente le cose vanno bene. Non è totalmente indolore fare una cache, non è così banale, infatti è stato un traguardo ingegneristico non banale, perché per esempio se avete un sistema multiprocessore, ognuno con la sua cache, occorre dei meccanismi, che i processori implementano, per invalidare la cache degli altri processori se uno modifica un dato nella cache, perché se no risulta dissolvinato... Una delle garanzie che deve avere una cache è che sia trasparente, che non si veda, che nessun programma abbia risultati diversi se eseguito in un sistema con la cache o senza.

E quindi siccome la semantics di accesso, l'adattazione nell'accesso alla memoria centrale è una semantics immediata, cioè mi aspetto che se un programma scrive a quella locazione un dato dal momento in cui è stato fatto l'operazione, tutti gli altri programmi leggono lo stesso dato, occorre che la cache garantisca questa cosa, e per farlo deve invalidare i contenuti della cache degli altri processori.

Come c'è una cache tra la memoria centrale e i registri, si può fare una cache dei contenuti dei dictioni nella RAM. Ancora una volta pensiamo la RAM ha velocità un milione di volte superiore a quella del disco.