Panoramica sui SO

Sistemi Operativi

Antonino Staiano
Email: antonino.staiano@uniparthenope.it

Introduzione

- Ambienti di calcolo e natura delle elaborazioni
- Classi di SO
- Efficienza, Prestazioni del sistema e servizi utente

- Sistemi di elaborazione a lotti (batch)
- Sistemi multi-programmati
- Sistemi time-sharing (a divisione di tempo)
- Sistemi Operativi real-time
- Sistemi Operativi distribuiti
- Sistemi Operativi moderni

Ambienti di Calcolo e Natura delle Elaborazioni

- Un ambiente di calcolo (o di elaborazione) consiste di
  - un computer,
  - le sue interfacce con altri sistemi,
  - i servizi forniti dal suo SO ai propri utenti ed i loro programmi
- Evoluzione
  - Ambienti di calcolo non interattivi
  - Ambienti di calcolo interattivi
  - Ambienti real-time, distribuiti e embedded
  - Ambienti di calcolo moderni

Ambienti di Calcolo e Natura delle Elaborazioni (cont.)

- Ambienti di calcolo non interattivi
  - Il SO è orientato sull'uso efficiente delle risorse
  - Elaborazioni sotto forma di programma o job
- Ambienti di calcolo interattivi
  - Il SO è orientato sulla riduzione della quantità media di tempo richiesto per implementare un'interazione tra un utente ed la propria elaborazione
  - L'esecuzione di un programma è chiamata processo

Ambienti di Calcolo e Natura delle Elaborazioni (cont.)

Elaborazioni in un SO

Computation	Description
Program	A *program* is a set of functions or modules, including some functions or modules obtained from libraries.
Job	A *job* is a sequence of programs that together achieve a common goal. It is not meaningful to execute a program in a job unless previous programs in the job have been executed successfully.
Process	A *process* is an execution of a program.
Subrequest	A *subrequest* is the presentation of a computational requirement by a user to a process. Each subrequest produces a single response, which consists of a set of results or actions.

Ambienti di Calcolo e Natura delle Elaborazioni (cont.)

- Ambienti real-time, distribuiti ed embedded
  - *Un'elaborazione real-time* ha specifici vincoli temporali
    - Il SO assicura che le elaborazioni siano completate nei vincoli
  - *Ambiente di calcolo distribuito*: permette un'elaborazione per usare risorse localizzate id più computer attraverso una rete
  - *Ambiente di calcolo embedded*: il computer è una parte di uno specifico sistema HW
    - Il computer è tipicamente poco costoso con una configurazione minimale
    - Il SO deve soddisfare i vincoli temporali che derivano dalla natura del sistema da controllare

Ambienti di Calcolo e Natura delle Elaborazioni (cont.)

- Ambienti di calcolo moderni
  - Supportano numerose e diversificate applicazioni
  - Hanno caratteristiche tratte dai diversi ambienti di calcolo appena descritti
    - Il SO impiega strategie complesse per gestire le elaborazioni dell'utente e le risorse
      - Ad esempio, deve ridurre il tempo medio richiesto per implementare un'interazione tra un utente e un'elaborazione e assicurare un uso efficiente delle risorse

Classi di SO

Caratteristiche chiave delle classi di SO

OS class	Period	Prime concern	Key concepts
Batch processing	1960s	CPU idle time	Automate transition between jobs
Multiprogramming	1960s	Resource utilization	Program priorities, preemption
Time-sharing	1970s	Good response time	Time slice, round-robin scheduling
Real time	1980s	Meeting time constraints	Real-time scheduling
Distributed	1990s	Resource sharing	Distributed control, transparency

Efficienza, Prestazioni del Sistema e Servizi Utente

- Due tra gli obiettivi fondamentali di un SO:
  - Efficienza d'uso
    - Di una risorsa
  - Convenienza per l'utente
    - Aspetto misurabile: servizi utente
      - Tempo di *turnaround*
      - Tempo di *risposta*
- Per un amministratore di sistema, sono più importanti le prestazioni di un sistema nel suo ambiente
  - Tipicamente misurate come *throughput*

Efficienza, Prestazioni del sistema e servizi utente (cont.)

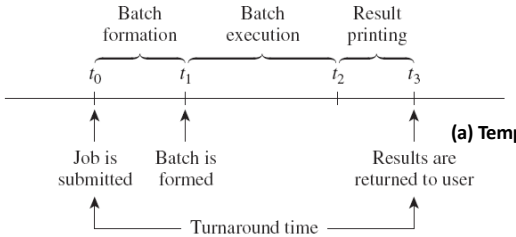
- **Definizione throughput**: il numero medio di job, programmi, processi o sotto-richieste completate da un sistema nell'unità di tempo
- **Definizione di turnaround**: Il tempo dalla sottomissione di un job, programma o processo da un utente fino all'istante in cui i risultati sono resi disponibili all'utente
- **Definizione di tempo di risposta**: il tempo dalla sottomissione di una sotto-richiesta dell'utente all'istante in cui un processo risponde ad essa

Aspect	Measure	Description
Efficiency of use	CPU efficiency	Percent utilization of the CPU
	Memory efficiency	Percent utilization of memory
System performance	Throughput	Amount of work done per unit time
User service	Turnaround time	Time to complete a job or a process
	Response time	Time to implement one subrequest

Sistemi di Elaborazione Batch

- *Batch*: *sequenza di job* utente preparati per essere elaborati dal SO
- In tal modo il kernel avvia l'elaborazione dei job senza che sia richiesto l'intervento dell'operatore al computer
- I lettori di schede e le stampanti costituivano un collo di bottiglia alle prestazioni degli anni '60
  - Per risolvere tale problema furono usati lettori di schede e stampanti virtuali realizzati mediante i nastri magnetici
- *Istruzioni di controllo* usate per proteggere contro le interferenze tra job
- Un interprete dei comandi (parte del kernel) legge una scheda quando il programma correntemente in esecuzione nel job richiede la prossima scheda

Sistemi di Elaborazione Batch (cont.)



(a) Tempo di *turnaround* di un job

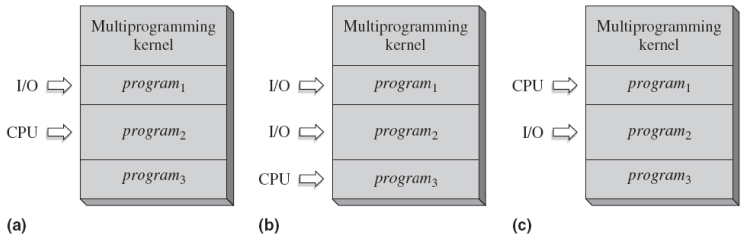
(b) Istruzioni di controllo IBM 360/370

```
// JOB ···          → "Start of job" statement
// EXEC FORTRAN     → Execute the Fortran compiler
                    } Fortran program
// EXEC             → Execute just compiled program
                    } Data for Fortran program
/*                  → "End of data" statement
/&                  → "End of job" statement
```

Sistemi Multi-Programmati

- Forniscono un utilizzo efficiente delle risorse in un ambiente non interattivo
- Usano la modalità DMA dell'I/O
  - Possono eseguire le operazioni di I/O di alcuni programmi mentre la CPU sta eseguendo qualche altro programma
    - Fa un uso efficiente sia della CPU che dei dispositivi di I/O
- In questi sistemi la misura più adatta per i **servizi utente** è il tempo di turnaround di un programma

Sistemi Multi-Programmati (cont.)



Funzionamento di un sistema multi-programmato:

(a) programma$_2$ è in esecuzione mentre programma$_1$ sta eseguendo un'operazione di I/O;

(b) programma$_2$ avvia un'operazione di I/O, programma$_3$ è schedulato;

(c) l'operazione di I/O del programma$_1$ è completata ed è schedulato.

Sistemi Multi-Programmati (cont.)

Feature	Description
DMA	The CPU initiates an I/O operation when an I/O instruction is executed. The DMA implements the data transfer involved in the I/O operation without involving the CPU and raises an I/O interrupt when the data transfer completes.
Memory protection	A program can access only the part of memory defined by contents of the *base register* and *size register*.
Kernel and user modes of CPU	Certain instructions, called *privileged instructions*, can be performed only when the CPU is in the kernel mode. A program interrupt is raised if a program tries to execute a privileged instruction when the CPU is in the user mode.

Sistemi Multi-Programmati (cont.)

- Una una **misura delle prestazioni** adeguata per un SO multi-programmato è il *throughput*
  - Rapporto tra il numero di programmi elaborati e il tempo totale necessario per elaborarli
- Il SO mantiene sempre un numero sufficiente di programmi in memoria in modo che la CPU e i dispositivi di memoria non siano inattivi (idle)
  - *Grado di multi-programmazione*: numero di programmi
  - Usa un giusto mix di programmi CPU-bound e I/O-bound
  - Assegna opportune priorità ai programmi CPU-bound e I/O-bound

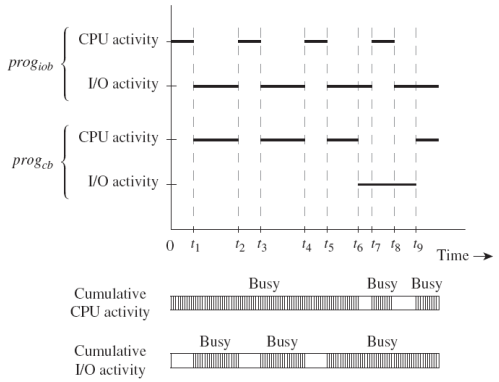
Priorità dei Programmi

Technique	Description
Appropriate *program mix*	The kernel keeps a mix of CPU-bound and I/O-bound programs in memory, where • A *CPU-bound program* is a program involving a lot of computation and very little I/O. It uses the CPU in long bursts—that is, it uses the CPU for a long time before starting an I/O operation. • An *I/O-bound program* involves very little computation and a lot of I/O. It uses the CPU in small bursts.
Priority-based preemptive scheduling	Every program is assigned a priority. The CPU is always allocated to the highest-priority program that wishes to use it. A low-priority program executing on the CPU is preempted if a higher-priority program wishes to use the CPU.

- **Definizione di priorità**: Un criterio di tie-break mediante il quale uno scheduler decide quale richiesta dovrebbe essere schedulata quando molte richieste sono in attesa.

Priorità dei Programmi (cont.)

In ambienti multi-programmati, un programma I/O-bound dovrebbe avere una priorità più alta rispetto ad un programma CPU-bound.

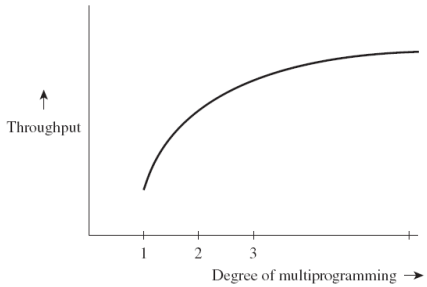


Prestazioni dei Sistemi Multi-Programmati

- Come migliorare le prestazioni?

Action	Effect
Add a CPU-bound program	A CPU-bound program (say, $prog_3$) can be introduced to utilize some of the CPU time that was wasted in Example 3.1 (e.g., the intervals t_6–t_7 and t_8–t_9). $prog_3$ would have the lowest priority. Hence its presence would not affect the progress of $prog_{cb}$ and $prog_{iob}$.
Add an I/O-bound program	An I/O-bound program (say, $prog_4$) can be introduced. Its priority would be between the priorities of $prog_{iob}$ and $prog_{cb}$. Presence of $prog_4$ would improve I/O utilization. It would not affect the progress of $prog_{iob}$ at all, since $prog_{iob}$ has the highest priority, and it would affect the progress of $prog_{cb}$ only marginally, since $prog_4$ does not use a significant amount of CPU time.

Prestazioni dei Sistemi Multi-Programmati (cont.)



Variazione del throughput rispetto al grado di multi-programmazione

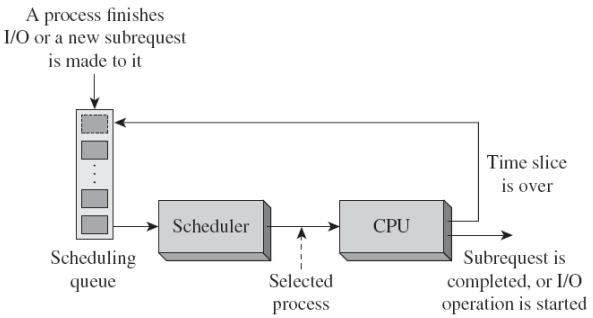
Quando viene mantenuto un appropriato mix di programmi, un aumento del grado di multi-programmazione incrementa il throughput.

Sistemi Time-sharing

- Fornisce una risposta veloce alle sotto-richieste dell'utente
  - *Scheduling round-robin con time-slicing*
    - *Il kernel mantiene una coda di scheduling*
    - Se il *time slice* scade prima che il processo completi il servizio di una sotto-richiesta, il kernel lo prelaziona, lo sposta alla fine della coda e schedula un altro processo
      - Implementato mediante un interrupt di un timer

Sistemi Time-sharing (cont.)

- **Definizione Time slice: l**a più grande quantità di tempo di CPU che un processo (time-shared) può consumare quando è schedulato per l'esecuzione.



Sistemi Time-sharing (cont.)

- Tempo di risposta (rt): misura dei **servizi utente**
  - Se l'elaborazione di una sotto-richiesta richiede δ secondi di CPU

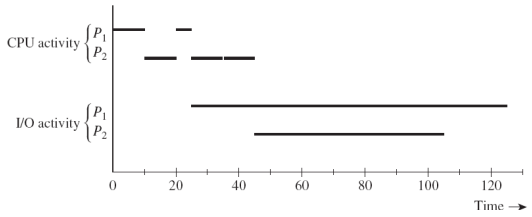
$$rt = n \times (\delta + \sigma)$$
$$\eta = \delta / (\delta + \sigma)$$

Dove η: efficienza di CPU,
σ: overhead dello scheduling,
n: numero utenti che usano il sistema,
δ: tempo richiesto per completare una sotto-richiesta

- Il tempo di risposta effettivo potrebbe essere diverso poiché
  - Alcuni utenti possono essere inattivi
  - Alcuni programmi possono richiedere più di δ secondi di CPU

Sistemi Time-sharing (cont.)

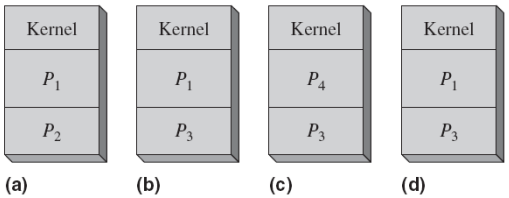
Time	Scheduling list	Scheduled program	Remarks
0	P_1, P_2	P_1	P_1 is preempted at 10 ms
10	P_2, P_1	P_2	P_2 is preempted at 20 ms
20	P_1, P_2	P_1	P_1 starts I/O at 25 ms
25	P_2	P_2	P_2 is preempted at 35 ms
35	P_2	P_2	P_2 starts I/O at 45 ms
45	–	–	CPU is idle



Funzionamento dei processi P1 e P2 in un sistema time-sharing

Swapping dei programmi

- **Swapping**: tecnica che rimuove temporaneamente un processo dalla memoria di un elaboratore
- Il kernel esegue operazioni di *swap-in* e *swap-out*



Swapping*:* (a) processi in memoria tra 0 e 105 ms; (b) P_2 è sostituto da P_3 a 105 ms; (c) P_1 è sostituito da P_4 a 125 ms; (d) P_1 è soggetto a swap-in per servire la successiva sotto-richiesta relativa ad esso.

Sistemi Operativi Real-time

- Nelle applicazioni real-time, gli utenti necessitano di computer per eseguire determinate azioni tempestivamente
  - Per controllare azioni in un sistema esterno o per prenderne parte
  - La puntualità dipende dai vincoli temporali
- **Applicazione real-time**: un programma che risponde ad attività in un sistema esterno, in un tempo massimo determinato dal sistema esterno
- Se l'applicazione impiega troppo tempo per rispondere ad un'attività, può verificarsi un fallimento nel sistema esterno
  - Requisito di risposta
  - Deadline: tempo entro cui dovrebbe essere eseguita un'azione

Sistemi Real-Time Soft e Hard

- Un sistema real-time hard soddisfa i requisiti di risposta in ogni condizione
  - Tipicamente è dedicato all'elaborazione di applicazioni real-time
- Un sistema real-time soft fa di tutto per soddisfare i requisiti di risposta di un'applicazione real-time
  - Non garantisce che ciò avvenga
    - Soddisfa i requisiti probabilisticamente
  - Ad esempio, applicazioni multimediali

Caratteristiche di un SO real-time

Feature	Explanation
Concurrency within an application	A programmer can indicate that some parts of an application should be executed concurrently with one another. The OS considers execution of each such part as a process.
Process priorities	A programmer can assign priorities to processes.
Scheduling	The OS uses priority-based or deadline-aware scheduling.
Domain-specific events, interrupts	A programmer can define special situations within the external system as events, associate interrupts with them, and specify event handling actions for them.
Predictability	Policies and overhead of the OS should be predictable.
Reliability	The OS ensures that an application can continue to function even when faults occur in the computer.

Sistemi Operativi Distribuiti

- Un sistema di computer distribuito consiste di diversi sistemi di computer singoli connessi attraverso una rete
  - Ciascun sistema di computer potrebbe essere un PC, un sistema multiprocessore o un cluster
  - In un sistema esistono molte tipologie di risorse
    - Questa caratteristica è usata per determinare una serie di benefici
  - Gestire i fallimenti di rete o di singoli sistemi richiede tecniche speciali
  - Gli utenti devono usare tecniche speciali per accedere alle risorse attraverso la rete

Sistemi Operativi Distribuiti (cont.)

Benefit	Description
Resource sharing	Resources can be utilized across boundaries of individual computer systems.
Reliability	The OS continues to function even when computer systems or resources in it fail.
Computation speedup	Processes of an application can be executed in different computer systems to speed up its completion.
Communication	Users can communicate among themselves irrespective of their locations in the system.

Tecniche Speciali per SO Distribuiti

- **Sistema distribuito**: Un sistema che consiste di due o più nodi, dove ogni nodo è un sistema di computer con i propri clock e memoria, hardware di rete e capacità di eseguire alcune funzioni di controllo di un SO

Concept/Technique	Description
Distributed control	A control function is performed through participation of several nodes, possibly *all* nodes, in a distributed system.
Transparency	A resource or service can be accessed without having to know its location in the distributed system.
Remote procedure call (RPC)	A process calls a procedure that is located in a different computer system. The RPC is analogous to a procedure or function call in a programming language, except that the OS passes parameters to the remote procedure over the network and returns its results over the network.

I Moderni SO

Concept	Typical example of use
Batch processing	To avoid time-consuming initializations for each use of a resource; e.g., database transactions are batch-processed in the back office and scientific computations are batch-processed in research organizations and clinical laboratories.
Priority-based preemptive scheduling	To provide a favored treatment to high-priority applications, and to achieve efficient use of resources by assigning high priorities to interactive processes and low priorities to noninteractive processes.
Time-slicing	To prevent a process from monopolizing the CPU; it helps in providing good response times.
Swapping	To increase the number of processes that can be serviced simultaneously; it helps in improving system performance and response times of processes.
Creating multiple processes in an application	To reduce the duration of an application; it is most effective when the application contains substantial CPU and I/O activities.
Resource sharing	To share resources such as laser printers or services such as file servers in a LAN environment.

Ricapitolando

- Un ambiente di calcolo consiste di un sistema di computer, le sue interfacce con altri sistemi e i servizi forniti dal suo SO ad utenti e programmi
  - Evoluto grazie allo sviluppo tecnologico in informatica
    - Sistemi di elaborazione batch
    - SO multiprogrammato
      - Scheduling basato su priorità
    - SO time-sharing
      - Scheduling round-robin con con slot temporali

Ricapitolando (cont.)

- Evoluzione (cont.)
  - SO real-time
    - Scheduling con priorità e scheduling basato su deadline
  - SO distribuiti
    - Consentono ai programmi di condividere risorse attraverso una rete di comunicazione
  - SO moderno
    - Un ambiente di elaborazione moderno contiene elementi di tutti gli ambienti di elaborazione classici
    - Usa tecniche diverse per applicazioni diverse