



Metodologie informatiche per l'organizzazione dei servizi turistici

(Prof. Nicola Orio)

Software

Corso di laurea in
Progettazione e gestione del turismo culturale



Programmazione

Scrivere software è un'attività complessa

- ✓ Formalizzare i compiti da svolgere
 - Si parla di scrittura di algoritmi
- ✓ Prevedere e gestire tutte le situazioni
 - Situazioni impreviste possono bloccare l'algoritmo
- ✓ Difficile fare test completi

Ci si appoggia a software già scritto

- ✓ Algoritmi complessi basati su algoritmi più semplici
 - Bisogna tenere traccia dei diversi componenti



Sistemi informativi

Ci sono diversi livelli di sistema informatico

- ✓ Dispositivo mobile
 - Un solo utente, controllo limitato
- ✓ Personal computer
 - Più utenti (non insieme), controllo quasi totale
- ✓ Server
 - Molti utenti insieme, un amministratore controlla
- ✓ Cloud
 - Molti computer, molti utenti, più amministratori



Tipologie di software

Per il computer i programmi servono

- ✓ A farlo funzionare
 - Senza questo software l'hardware è inutile
 - Essendo così importante è già installato all'acquisto
- ✓ A specializzarlo per svolgere compiti particolari
 - Renderlo uno strumento di lavoro/svago
 - Alcuni programmi sono già installati, per comodità
 - Ogni utente aggiunge gli altri programmi a sua scelta

Per il processore non c'è differenza

- Sono in tutti i casi delle sequenze di operazioni



Sistema operativo – 1

Consente l'uso del computer

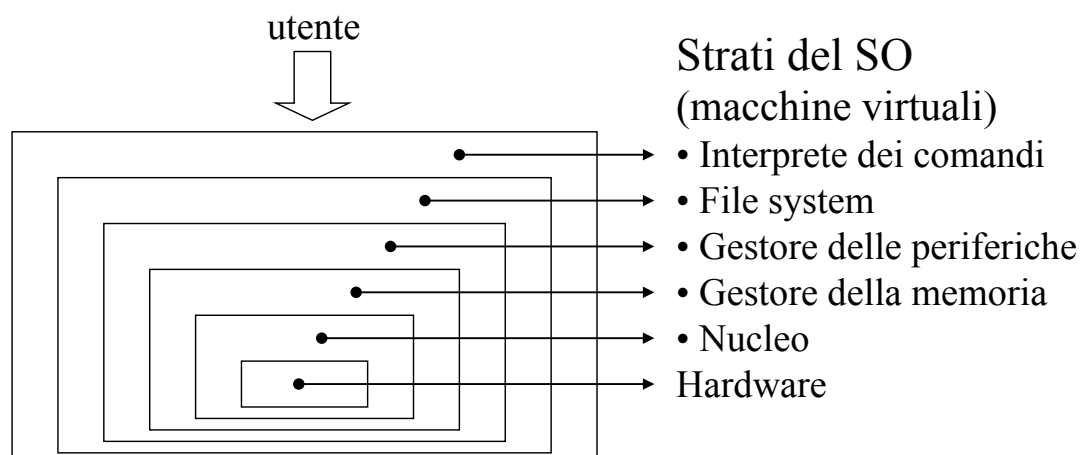
- ✓ Insieme di programmi che
 - Controlla e sincronizza i programmi attivi
 - Gestisce la memoria centrale
 - Consente la comunicazione con le periferiche
 - Gestisce i dati in memoria di massa
 - Interpreta le istruzioni dell'utente
- ✓ Gli errori si riflettono su tutto il sistema
 - Pochi sistemi operativi in commercio



Sistema operativo – 2

Organizzato in più strati che

- ✓ Implementano una funzionalità
- ✓ Comunicano prevalentemente con gli strati vicini





Interprete dei comandi

Strato più esterno per comunicare con l'utente

- ✓ Interazione testuale
 - Istruzioni in forma di comandi utilizzando la tastiera
 - Bisogna ricordare nome e sintassi dei comandi
 - Si possono eseguire operazioni complesse
- ✓ Interazione grafica
 - Funzioni rappresentate graficamente da icone
 - Selezione con interfaccia di puntamento
 - Bisogna selezionare ogni volta gli oggetti e i comandi da dare
 - Invece di ricordare i comandi si devono riconoscere le icone



File System – 1

Si occupa della gestione dei file, che

- ✓ Sono porzioni della memoria di massa, con un nome
 - Di solito diviso in due parti
 - Nome vero e proprio: sequenza di caratteri
 - Estensione: un punto seguito da uno o più caratteri
 - Creato dall'utente o dai programmi stessi
 - Appiglio mnemonico per ipotizzare il contenuto del file
 - La cui estensione indica
 - Come sono organizzati i dati contenuti nei file
 - Il tipo di programma in grado di interpretare il file



File System – 2

I file risiedono in memoria di massa

- ✓ La struttura fisica del supporto è trasparente
- ✓ Si possono creare dei supporti virtuali
 - Non corrispondono a degli effettivi supporti fisici
- ✓ Lo spazio è gestito automaticamente
 - Ricerca di spazio libero per nuovi file
 - Cancellazione rapida

Sistemi multiutente

- ✓ Gestione del controllo degli accessi



File System – 3

Fornisce dei metodi di uso generale

- ✓ Leggono, scrivono, cancellano, aggiornano i file
- ✓ Possono essere invocati da
 - Utente tramite l'interprete dei comandi
 - Applicazioni che gestiscono documenti
 - Anche altri strati del sistema operativo
- ✓ Accede ai file specificandone il nome
 - Il nome di un file dovrebbe essere unico
 - scomodo per gli utenti
 - Problemi con tanti file diversi e di diversa provenienza



Directory – 1

Definizione di directory

- ✓ Struttura virtuale gestita dal file system che può
 - Essere vuota
 - Contenere uno o più file
 - Possibili limitazioni al numero di file
 - Contenere una o più directory
 - Possibili limitazioni alla profondità di sub-directory

Ha la funzione di organizzare i file

- ✓ Utile per l'utente e per molte applicazioni



Directory – 2

Struttura ad albero rovesciato

- Directory radice (root)
 - Non è contenuta in nessun'altra directory
- Altre directory sono i rami
- I file sono le foglie
- ✓ Unico percorso per raggiungere il file
 - Chiamato pathname
 - Pathname = posizione + nome

Esempio (stile web): /user/orio/documenti/unipd/software.pptx

- File con posizione diversa possono avere lo stesso nome



Installazione

Per installare nuovo software bisogna

- ✓ Copiare il codice in memoria di massa
 - Creare nuovi file e directory utili al software
 - Creare le icone associate
- ✓ Notificare la presenza al sistema operativo
 - Il software può avere funzionalità utili anche ad altri software

Spesso gestita da programmi detti wizard

- ✓ Attenzione a cosa viene installato
- ✓ Alcuni programmi sono venduti preinstallati nel SO



Avvio

Le applicazioni vengono avviate da

- ✓ Utente tramite l'interprete dei comandi
- ✓ Altre applicazioni (anche dal sistema operativo)

All'avvio

- Il sistema operativo
 - Carica in memoria centrale il file dell'applicazione
- La CPU
 - Esegue le istruzioni elementari contenute nel file
 - Non fa nessun controllo sulla correttezza del software
- L'utente
 - Deve fidarsi che tutto vada a buon fine e che non sia malware



Disinstallazione

Quando un'applicazione non interessa più

- ✓ Cancellare il software non è sufficiente
- ✓ Bisogna notificare la rimozione al sistema operativo
 - Vengono cancellati anche i file di utilità nel sistema

Operazione non facile per utenti inesperti

- ✓ Molti sistemi operativi hanno strumenti adeguati
 - Gestione applicazioni in Windows
 - Cancellazione più facile per Mac OS X, Android, iOS
- ✓ Cancellazione guidata per alcuni programmi
 - Ai programmatori conviene sia macchinoso



Codice macchina

Il processore esegue operazioni molto semplici

- ✓ I programmi sono costituiti da milioni istruzioni elementari
 - Il processore è rapidissimo ad eseguirle
 - L'utente non ha la percezione di quanto siano elementari

Problema: troppe istruzioni, troppo semplici

- ✓ E' quasi impossibile capire cosa farà un programma leggendo il codice macchina
- ✓ I programmatori scrivono in codice macchina in casi rarissimi, quando l'efficienza è fondamentale



Codice sorgente

I programmatori usano linguaggi ad alto livello

- ✓ Conversione automatica in linguaggio macchina
 - Eseguita da software detti compilatori e interpreti
 - Da codice sorgente (linguaggio del programmatore)
 - A codice macchina (linguaggio del processore)
 - Il processo non è invertibile
 - Dal codice macchina è quasi possibile risalire al codice sorgente
 - Modifiche, correzioni, aggiornamenti sono fatte sul codice sorgente
 - Viene generato e distribuito un nuovo codice macchina



Open source – 1

Con l'open source si rilascia il codice sorgente

- ✓ Per funzionare basta il codice macchina
- ✓ Se altri programmatori accedono al codice sorgente
 - Possono copiarlo e/o rivenderlo
 - I propri “trucchi” diventano noti
 - Le imperfezioni o le parti mancanti sono visibili
 - Possono aggiornarlo per conto proprio
 - Non ci si deve rivolgere al programmatore iniziale
 - Perdita del controllo del proprio software
 - Le grandi software-house non amano l'open source



Open source – 2

Motivazioni a favore

- ✓ Per i programmatori
 - Collaborazione per software efficiente e sicuro
 - Molte parti già disponibili da adattare alle proprie esigenze
 - Alternativa al potere delle grandi software house
- ✓ Per gli utenti
 - Sicurezza e affidabilità
 - Si può leggere cosa fa un programma
 - Indipendenza
 - Si può chiedere a terzi di fare modifiche
 - Personalizzazione



Creative Commons

Concetto di open source applicato ai documenti

- ✓ Applicato a tutti i media
 - Musica, fotografie, video, testi
- ✓ Diversi gradi di licenza
 - Dall'utilizzo in qualsiasi forma
 - Anche commerciale, anche di parti
 - Al solo ascolto
 - Solo integrare e per uso personale

Concetti mutuati da open source e shareware



Trusted computing

Nato per evitare intercettazioni e manomissioni

- ✓ Hardware e software appositi
 - Strumenti principali
 - Cifratura del software (endorsement key)
 - Controllo periferiche (secure I/O)
 - Memorie separate (memory curtaining)
 - Memoria bloccata (sealed storage)
 - Controllo remoto dello stato (remote attestation)
- ✓ Poco noto ma supportato dalle multinazionali
 - Diminuzione del controllo da parte degli utenti
 - Difficile bilanciamento tra sicurezza e privacy