

SHELL -> INTERPRETE DI COMANDI

KERNEL È IL NUCLEO E L'UTENTE PER INTERAGIRE INDIRETTAMENTE CON ESSO USA UNA SHELL (GUSCIO INTORNO). SI INTERAGISCE CON LA SHELL TRAMITE **TERMINALE**

IL TERMINALE È UN'EVOLOUZIONE DELLA TELESCHIVENTE, COMPOSTO DA UNA TASTIERA E UN DISPOSITIVO DI OUTPUT PER VEDERE IL RISULTATO

SU LINUX CI SONO DUE CONCETTI SIMILI :

- LE VIRTUAL CONSOLE, DEI TERMINALI CHE CONDIVIDONO TASTIERA E SCHERMO, E SI PUÒ PASSARE DA UNA ALL'ALTRA CON DELLE COMBINAZIONI DI TASTI (NON MOLTO USATI)
- EMULATORE DI TERMINALE, IMPLEMENTATI TRAMITE PSEUDOTERMINALI, OSSIA FILE CHE FINGONO DI ESSERE UN TERMINALE TRAMITE PROGRAMMA

-

OGNI PROCESSO SI TROVA DI DEFAULT 3 **FILE DESCRIPTOR** -> NUMERO INTERO CHE RAPPRESENTA UN FILE APERTO PER UN PROCESSO

0 : STANDARD INPUT (CIN IN C++)

1 : STANDARD OUTPUT (COUT IN C++)

2 : STANDARD ERROR (CERR IN C++)

OGNI PROCESSO HA UNA **CARTELLA DI LAVORO (DIRECTORY)**

IL FILE SYSTEM È UN ALBERO CHE CONTIENE DIRECTORY, CHE CONTENGONO DIRECTORY CHE CONTENGONO FILE ECC

CI SONO ALCUNE **DIRECTORY STANDARD**, QUINDI IN OGNI SISTEMA UNIX TROVIAMO:

- / : RADICE
- /BIN e /SBIN : COMANDI ESSENZIALI E COMANDI PER AMMINISTRARE IL SISTEMA
- /BOOT : FILE PER IL BOOT DEL SISTEMA
- /DEV : FILE SPECIALI CHE CORRISPONDONO A DISPOSITIVI
- /ETC : FILE PER LA CONFIGURAZIONE DEL SISTEMA
- /HOME e /ROOT : CARTELLE HOME (CARTELLE INIZIALI) PER GLI UTENTI E PER ROOT (OGNI UTENTE HA UNA CARTELLA NELLA HOME PER ESEMPIO L'UTENTE PIPPO È IN /HOME/PIPP0)
- /LIB* : LIBRERIE
- /MEDIA e /MNT : MOUNT-POINT (QUANDO IN UN SISTEMA UNIX VOGLIAMO USARE UN ALTRO DISCO, UNA USB ECC, PER POTERNE VEDERE IL CONTENUTO DOVETE MONTARE IL FILE SYSTEM

Della USB... IN UN NODO DELL'ALBERO CHE AVETE GIÀ, OSSIA L'UNICO ALBERO DEL FILE SYSTEM. IN UN WINDOWS INVECE GLI ALBERI SONO DIVISI, CON C PRIMO DISCO DI WINDOWS)

- /PROC e /SYS : FILE SYSTEM VIRTUALI
- /TMP : FILE TEMPORANEI
- /USR : GERARCHIA SECONDARIA, QUEI FILE CHE POSSONO ESSERE MONTATI IN SOLA LETTURA

TRA IL TERMINALE E I PROCESSI, C'È LA DISCIPLINA DI LINEA: ESEMPIO QUANDO SCRIVETE UN PROGRAMMA CHE LEGGE UN INPUT, NON VOLETE GESTIRE ANCHE IL FATTO CHE L'UTENTE CANCELLA UN CARATTERE CON BACKSPACE ECC, È UN LAVORO DELLA DISCIPLINA DI LINEA -> QUANDO UN PROGRAMMA LEGGE DA STANDARD INPUT, QUELLO CHE RICEVE È SOLO IL RISULTATO FINALE

LA SHELL IN MODALITÀ INTERATTIVA, STAMPA IL PROMPT DI COMANDI (UNA SEQUENZA DI CARATTERI CHE INDICA CHE È IN ATTESA DI COMANDI)

LA BESH LEGGE UNA LINEA DI INPUT E DOPODICHÉ:

- SEPARA IN TOKEN (ANALISI LESSICALE): PAROLE E OPERATORI
- ESPANDE GLI ALIAS -> DELLE STRINGHE CHE VENGONO ESPANSE IN MANIERA PIÙ ELABORATA
- FA IL PARSING DI COMANDI SEMPLICI E COMANDI COMPOSTI
- FA DELLE ESPANSIONI (ESEMPIO a* VIENE SOSTITUITO DA TUTTI I FILE CHE INIZIANO CON a, QUINDI "QUALSIASI SEQUENZA DI CARATTERI ANCHE VUOTA" (LPO))
- ESEGUE LE VARIE RIDIREZIONI DI INPUT E OUTPUT
- ESEGUE IL COMANDO (CHE POSSONO ESSERE DI TIPO BUILT-IN, OSSIA IMPLEMENTATI DIRETTAMENTE DALLA BESH, E TUTTI GLI ALTRI, OSSIA COMANDI ESTERNI)

I COMANDI PIÙ IMPORTANTI SONO:

- MAN : PER CERCARE NEL MANUALE (PER I COMANDI ESTERNI)
- HELP : PER CERCARE AIUTO NEI COMANDI BUILT-IN