



I meno giovani ricorderanno che quando si accendeva un Olivetti M24, dopo qualche minuto, dopo aver eseguito il POST, (il programma residente nel BIOS), letto ed eseguito quanto specificato nei file autoexec.bat e config.sys, appariva sullo schermo il prompt del DOS C:/> di colore verde su sfondo nero e nient'altro.



Tutto quello che si poteva fare lo si doveva fare tramite la riga di comando ovvero si dovevano impartire dei comandi al sistema operativo DOS scrivendoli dove lampeggiava il cursore.



Analogamente, quando si accendeva una macchina UNIX, (molto tempo prima del DOS), dopo una serie di indicazioni sullo schermo (alquanto esoteriche) si otteneva uno schermo nero con un prompt bianco e il cursore lampeggiante in attesa di ricevere comandi. Questa era la shell standard di unix (/bin/sh) chiamata anche Bourne shell dal nome del suo autore originario Stephen Bourne.



Anche quando si accende un server GNU/Linux (che normalmente non ha interfaccia grafica), dopo qualche minuto appare un prompt bianco su uno sfondo nero e il cursore lampeggiante "nome_utente@nome_computer:~\$".



Questo interprete dei comandi si chiama BASH che è l'acronimo di Bourne again shell.

Altre shell alternative a bash e piuttosto diffuse sono la Korn shell, la C shell e tcsh.

Quando invece si accende un computer desktop con sistema operativo GNU/Linux, per esempio Ubuntu, la schermata finale è analoga a quella di Windows, un desktop grafico, più o meno colorato, con le icone e tutto il resto.

Allora, questo significa che la Shell è presente solo sui server e non sui desktop?

La tipologia della schermata finale che si ha alla fine del Boot dipende da come è settato il runlevel.



Il runlevel rappresenta lo stato di attività di una macchina (uno di sette stati), relativamente ai programmi in esecuzione e ai servizi offerti.

Questo stato è identificato da un numero che va da 0 a 6:

0 corrisponde a spegnere la macchina

1 avvia la macchina in modalità singolo utente

2 per la modalità testo multiutente

3 come la 2 ma con il supporto di rete

4 in genere inutilizzato o dedicato a compiti specifici

5 per la modalità grafica multiutente con supporto della rete

6 non fa altro che riavviare il sistema



Quindi un server Ubuntu ha un valore di runlevel pari a 3 (modalità testo multiutente con supporto di rete) (per passare dallo stato 3 allo stato 5 è sufficiente digitare il comando startx a condizione che sia stato installato in precedenza il server grafico X) mentre un desktop Ubuntu ha un runlevel pari a 5 (modalità grafica multiutente con supporto di rete)

Quando la macchina ha un runlevel settato a 5, si può accedere alla Shell di Bash ovvero tornare a 3?

Si può accedere alla shell bash in due distinte modalità : console e terminale

La console è puramente testuale e a schermo intero, si può accedere a 6 diverse console premendo la combinazione di tasti

Ctrl+Alt+F1 ----> tty1 teletypewriter1 (terminale 1)

Ctrl+Alt+F2 ----> tty2 teletypewriter2 (terminale 2)

Ctrl+Alt+F3 ----> tty3 teletypewriter3 (terminale 3)

Ctrl+Alt+F4 ----> tty4 teletypewriter4 (terminale 4)

Ctrl+Alt+F5 ----> tty5 teletypewriter5 (terminale 5)

Ctrl+Alt+F6 ----> tty6 teletypewriter6 (terminale 6)

Ctrl+Alt+F7 chiude la console attiva e riappare il 1° desktop grafico se esiste

Ctrl+Alt+F8 chiude la console attiva e riappare il 2° desktop grafico se esiste

Ctrl+Alt+F9 chiude la console attiva e riappare il 3° desktop grafico se esiste

Ctrl+Alt+F10 chiude la console attiva e riappare il 4° desktop grafico se esiste

Ctrl+Alt+F11 chiude la console attiva e riappare il 5° desktop grafico se esiste

Se invece non si vuole uscire dall'ambiente grafico si possono aprire n terminali che altro non sono che console che girano dentro delle finestre.

PERCHE' UTILIZZARE LA SHELL (console o terminali)??

Ufficialmente la risposta è che tramite la riga di comando si riesce a fare molto di più e più velocemente rispetto a quanto si riesca a fare tramite i programmi grafici, ma questa, a mio parere, è una reminiscenza da sistemista.



I sistemisti (GNU/Linux ora e UNIX allora) operando su server senza interfaccia grafica avevano la console testuale come unico strumento di amministrazione. Di solito si collegano da remoto ad una macchina Windows dalla quale tramite una sessione telnet (Putty) accedono al server Linux con una console (terminale).

Un'altra risposta ufficiale è che "tutto quello che si può fare graficamente si può anche fare dalla riga di comando ma non il viceversa".

Se tutto questo è vero per i sistemisti che amministrano server, non è affatto detto che un semplice utilizzatore di Ubuntu in versione Desktop debba necessariamente conoscere tutti i comandi Bash, tutte le migliaia di opzioni e il linguaggio di scripting.

Anzi, questo, probabilmente è quello che ha impedito finora la diffusione di GNU/Linux a livello domestico.



L'utente (non sistemista) può sopravvivere benissimo "graficamente" e ricorre al terminale in poche circostanze e specificatamente quando ha necessità di diventare SuperUtente (root) per installare/disinstallare programmi e per modificare file e directory riservate all'amministratore root.

In questi casi specifici è sufficiente aprire un terminale e anteporre sudo al comando o al programma necessario.



Per esempio sudo nautilus, sudo gedit, ecc..

Un altro motivo per usare/conoscere la riga di comando è per esempio quando il server grafico, per qualche strana ragione, si blocca e non si riesce nemmeno a spegnere la macchina normalmente (non togliendo l'alimentazione).

In questo caso è sempre possibile aprire una console (runlevel 3) e digitare "sudo shutdown -h now)



Oppure per "killare" una finestra che non risponde si apre con Alt+F2 "l'equivalente di esegui" e si può invocare il programma xkill.

E' Importante precisare che il sistema Linux è CASE SENSITIVE, ossia distingue tra maiuscole e minuscole.

Spesso errori nello specificare un percorso o il nome di uno strumento sono dovuti proprio a questo.

E' opportuno quindi utilizzare il comando di completamento automatico [Tab].

Comandi DOS e loro equivalenti UNIX-GNU/Linux

ASSIGN	ln	collega file o directory
ATTRIB	chmod	cambia i permessi del file
CD	cd	cambia directory
CHDIR	cd	cambia directory
CLS	clear	pulisce lo schermo
COMP	diff, comm, cmp	confronta i file
COPY	cp	copia i file
Ctl-C	Ctl-C	interruzione (segnale)
Ctl-Z	Ctl-D	EOF (end-of-file)
DEL	rm	cancella il/i file
DELTREE	rm -rf	cancella ricorsivamente una directory
DIR	ls -l	elenca una directory
ERASE	rm	cancella il/i file
EXIT	exit	esce dal processo corrente
FC	comm, cmp	confronta i file
FIND	grep	ricerca le stringhe nei file
MD	mkdir	crea una directory
MKDIR	mkdir	crea una directory
MORE	more	filtro per l'impaginazione del testo del file
MOVE	mv	spostamento
PATH	\$PATH	percorso degli eseguibili
REN	mv	rinomina (sposta)
RENAME	mv	rinomina (sposta)
RD	rmdir	cancella una directory
RMDIR	rmdir	cancella una directory
SORT	sort	ordina il file
TIME	date	visualizza l'ora di sistema
TYPE	cat	visualizza il file allo stdout
XCOPY	cp	copia (estesa) di file

webmaster@comunecampagnano.it

[Invia questa pagina per email](#) [Salva come PDF](#)  [HOME](#)

<http://augustoscato lini.tk> <http://miniguide.tk> <http://linuxglassbell.sourceforge.net> [amministrazione](#)
<http://linuxbasic.altervista.org> augascaATyahoo.it