

BASH: Bourne Again Shell



Linux Day 2006

Goffredo Baroncelli <kreijack@inwind.it>

gpg key@ keyserver.linux.it: Goffredo Baroncelli (ghigo) <kreijack@inwind.it>

Key fingerprint = CE3C 7E01 6782 30A3 5B87 87C0 BB86 505C 6B2A CFF9

Indice

Attualità della shell

Cosa si intende per una shell

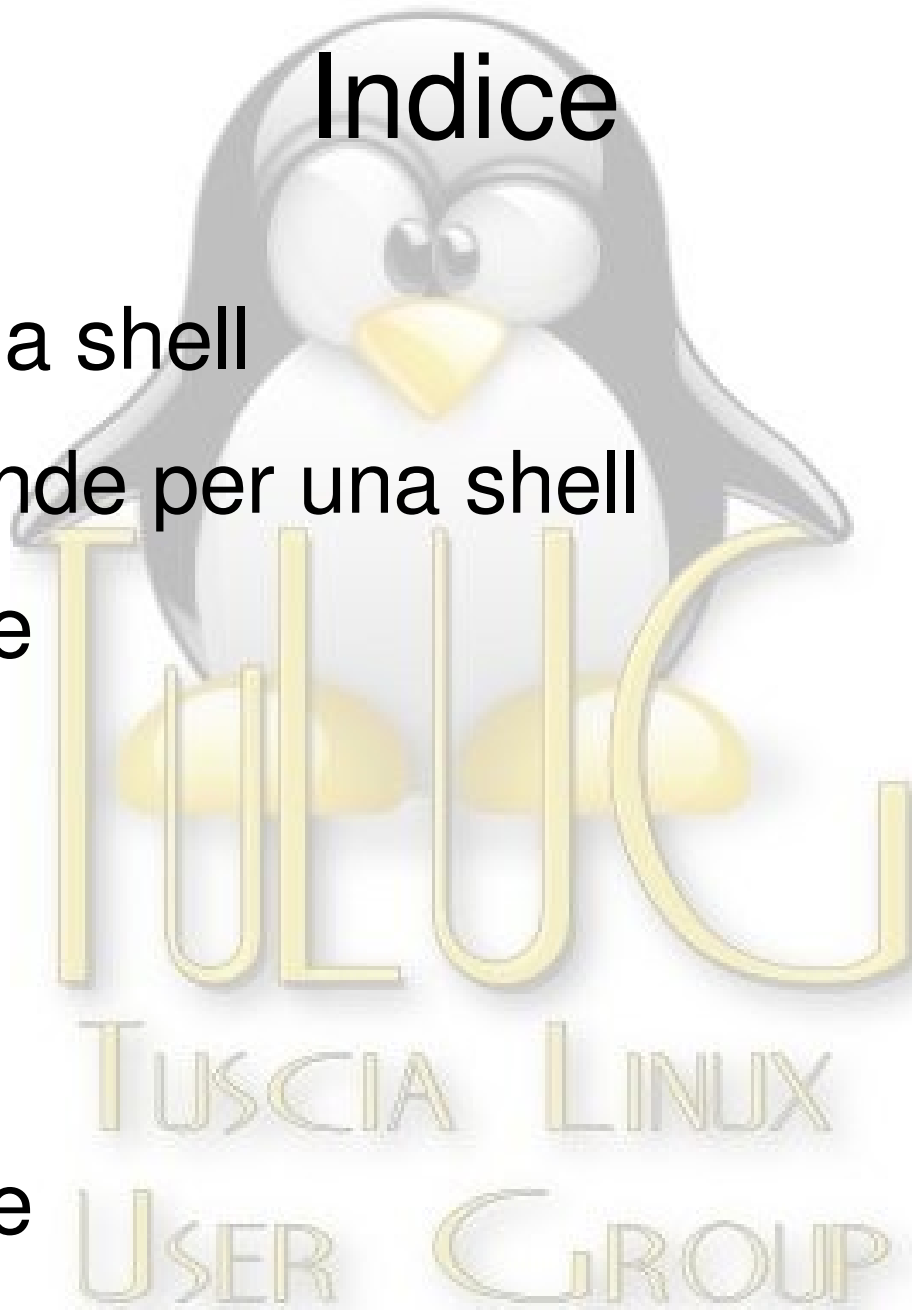
Esempi base

Piping

Job

Scripting

Uso variabile



Attualità della *shell*

- Unix nacque negli anni '70, e a quel tempo non c'erano interfacce grafiche
- Spesso le uniche interfacce erano le schede perforate e le stampanti
- Poi vennero i terminali seriali, e nacque l'esigenza di avere una shell interattiva (CLI)

TUSCIA LINUX
USER GROUP

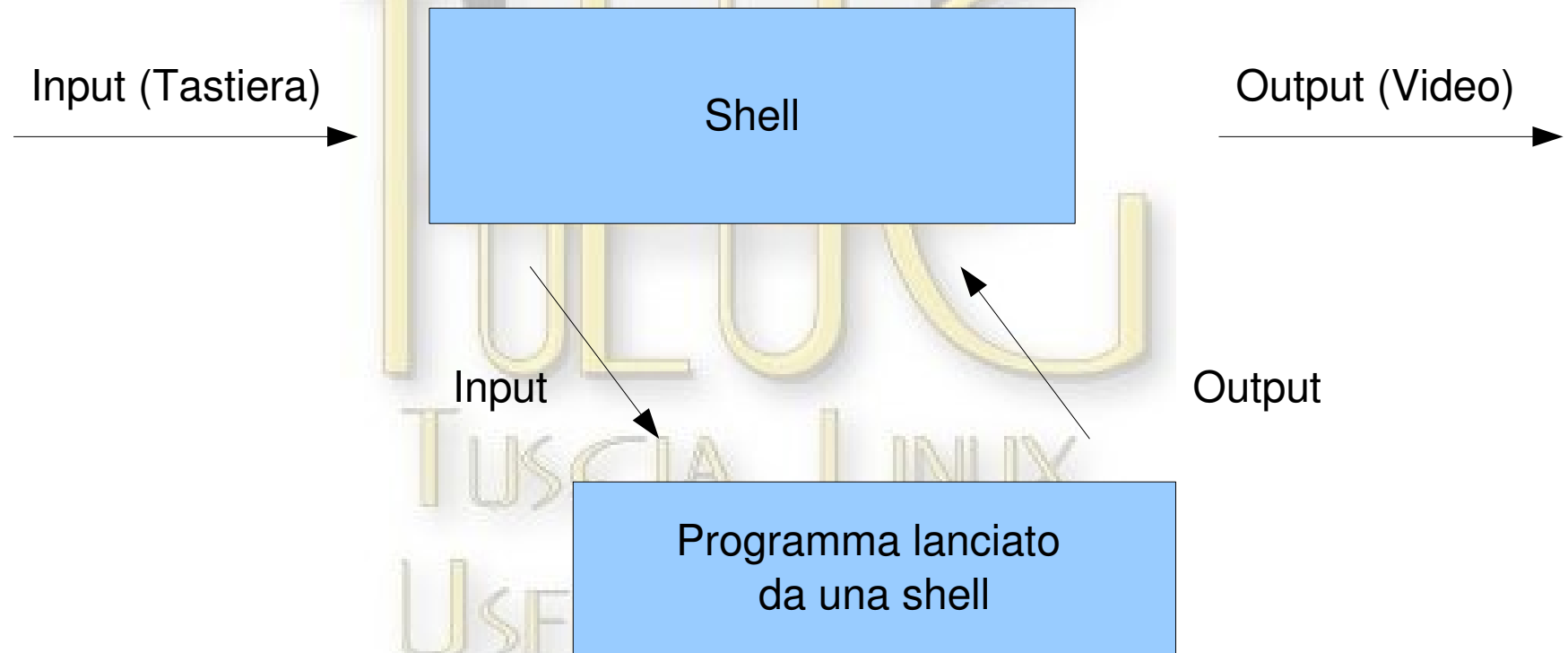
Attualità della *shell*

Vantaggi

- Limitato uso di banda
- Possibilità di scripting
- Creazione di processi complessi partendo da programmi semplici
- Facilità d'uso in caso di operazioni ripetute

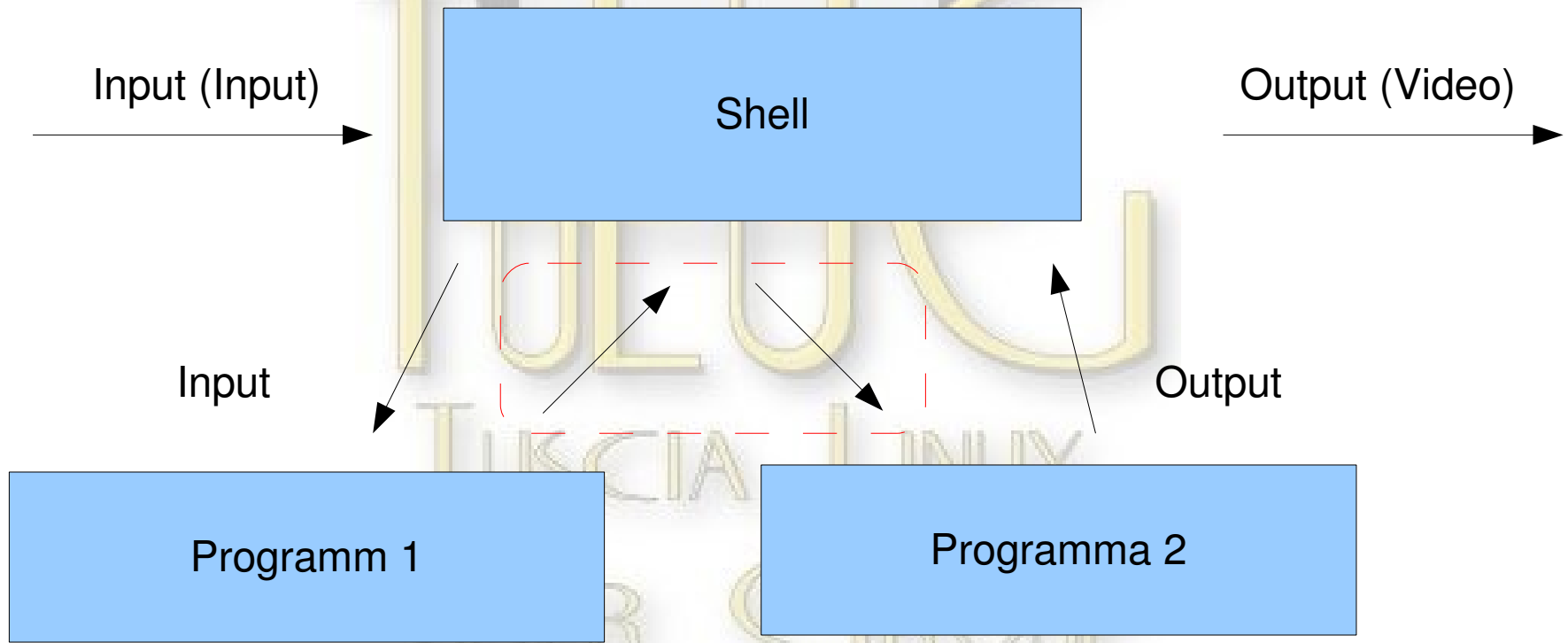
Funzionamento di una *shell*

Un programma tipicamente ha un *input* ed un *output*, che vengono gestiti dalla shell



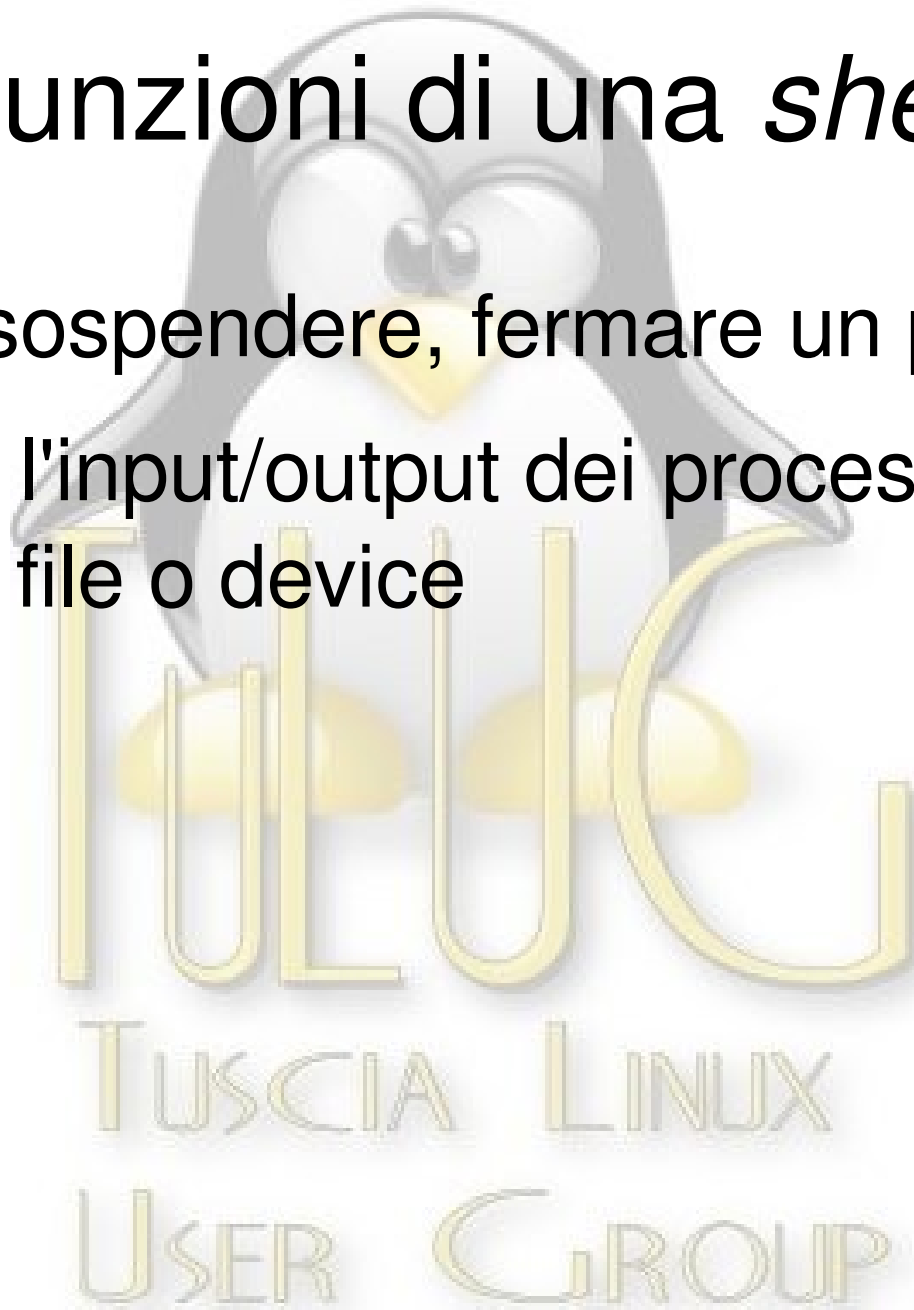
Funzionamento di una *shell*

Una shell può lanciare più programmi e gestire i loro input output



Funzioni di una *shell*

- Lanciare, sospendere, fermare un processo
- Redirigere l'input/output dei processi verso altri processi o file o device
- Scripting
- [...]



Esempi di comandi di una *shell*

```
utente$ ls
```

dira dirb
filea fileb
[...]

Prompt

Comando

Output

```
utente$ cat /etc/passwd  
root:x:0:0:root:/root:/bin/bash  
daemon:x:1:1:daemon:/usr/sbin:/bin/s  
[...]
```

Esempi di comandi di una *shell*

```
utente$ ls >/tmp/file &
```

Simbolo di
re-direzione

Manda il
processo in
background

Come facciamo a sapere quando ha finito ?

```
[2] 5933
```

PID processo

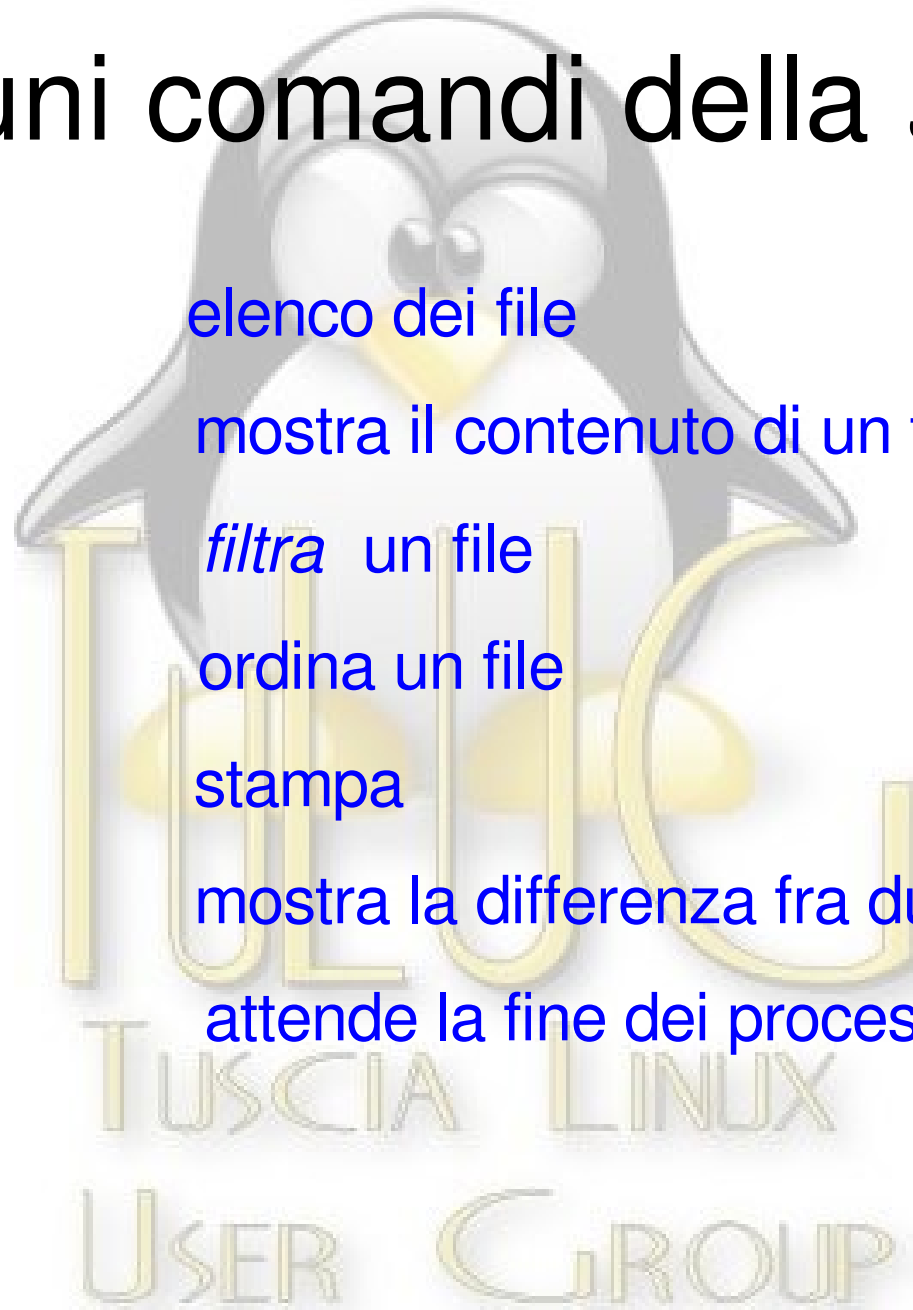
```
[2]+ Done
```

```
ls >/tmp/file
```

Fine processo

Alcuni comandi della *shell*

- ls, find
elenco dei file
- cat
mostra il contenuto di un file
- grep
filtra un file
- sort
ordina un file
- echo
stampa
- diff
mostra la differenza fra due file
- wait
attende la fine dei processi in background



Alcuni esempi

Elenco ordinato di file aventi nel nome la parola
fattura

```
utente$ find /mnt/disco | sort |  
grep fattura >/tmp/elenco-fatture &
```

TUSCIA LINUX
USER GROUP

Alcuni esempi

Differenza di due directory

```
utente$ ( find /mnt/disco | sort >disco.lst &  
find /mnt/backup | sort >backup.lst &  
wait;  
diff backup.lst disco.lst ) >diff.lst &
```

TUSCIA LINUX
USER GROUP

Scripting



- Uno dei grandi vantaggi delle shell lo scripting
- Lo startup dei sistemi Unix normalmente è fatto attraverso script
 - Rilevamento dell'hardware
 - Configurazione rete
 - Configurazione device
 - Partenza dei servizi
 - [...]

Scripting: un esempio



```
utente$ cat delete
```

```
# /bin/sh
```

```
mkdir /tmp/trash
```

```
mv $1 /tmp/trash
```

```
utente$ ls pippo
```

```
pippo
```

```
utente$ delete pippo
```

```
utente$ ls /tmp/trash
```

```
pippo
```

Scripting: un esempio (2)

```
# /bin/sh  
  
find /mnt/disco | sort | \  
    grep fattura >/tmp/elenco-fatture  
  
ls -l $( cat /tmp/elenco-fatture )
```

TUSCIA LINUX

USER GROUP

Variabili



Come tutti i linguaggi, anche la shell dispone di variabili. Le variabili (definite d'ambiente) possono anche essere utilizzate per settare alcune caratteristiche della shell (esempi).

Le variabili d'ambiente sono ereditate nei processi figli. Quindi possono essere usate per comunicare informazioni ai processi figli (esempi: DISPLAY, PATH...).

Variabili (2)

Esempio uso di una variabile

```
tulug:~$ A=123  
tulug:~$ echo $A  
123
```

Esempio configurazione

```
tulug:~$ echo $PS1  
\u:\w$  
tulug:~$ PS1="ciccio:\w\$ "
```

Esempio uso \$() [O backtick `]

```
ciccio:~$ A=$(ls)  
ciccio:~$ echo $A  
Desktop Examples
```



Domande ?

Bibliografia



Sito GNU: <http://www.gnu.org/software/bash/>

FAQ: <ftp://ftp.cwru.edu/pub/bash/FAQ>

Slide: <http://goffredo-baroncelli.homelinux.net/slide/linux-day.odp>

Wikipedia:

http://en.wikipedia.org/wiki/Bourne_shell,

<http://en.wikipedia.org/wiki/Bash>

[http://en.wikipedia.org/wiki/Shell_\(computing\)](http://en.wikipedia.org/wiki/Shell_(computing))