



UNIVERSITAT POLITÈCNICA DE CATALUNYA
BARCELONATECH

Departament d'Arquitectura de Computadors



**Barcelona
Supercomputing
Center**

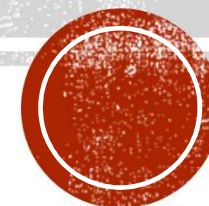
Centro Nacional de Supercomputación

EN EL PRINCIPI FOU EL SHELL

Shell workshop, El Club de les Perforistes Mortes

Xavier Martorell

27 de setembre de 2023



Echo “Què farem avui?”

- [12:05] Benvinguda
- Introducció
- Entorn d'execució
- Intèrpret de comandes
 - Execució de comandes
 - Variables i expressions
 - Estructures de control: seqüència, condicional i bucle
- Sortida al terminal
- Resum i exercici
- [~13:00] Final de la sessió

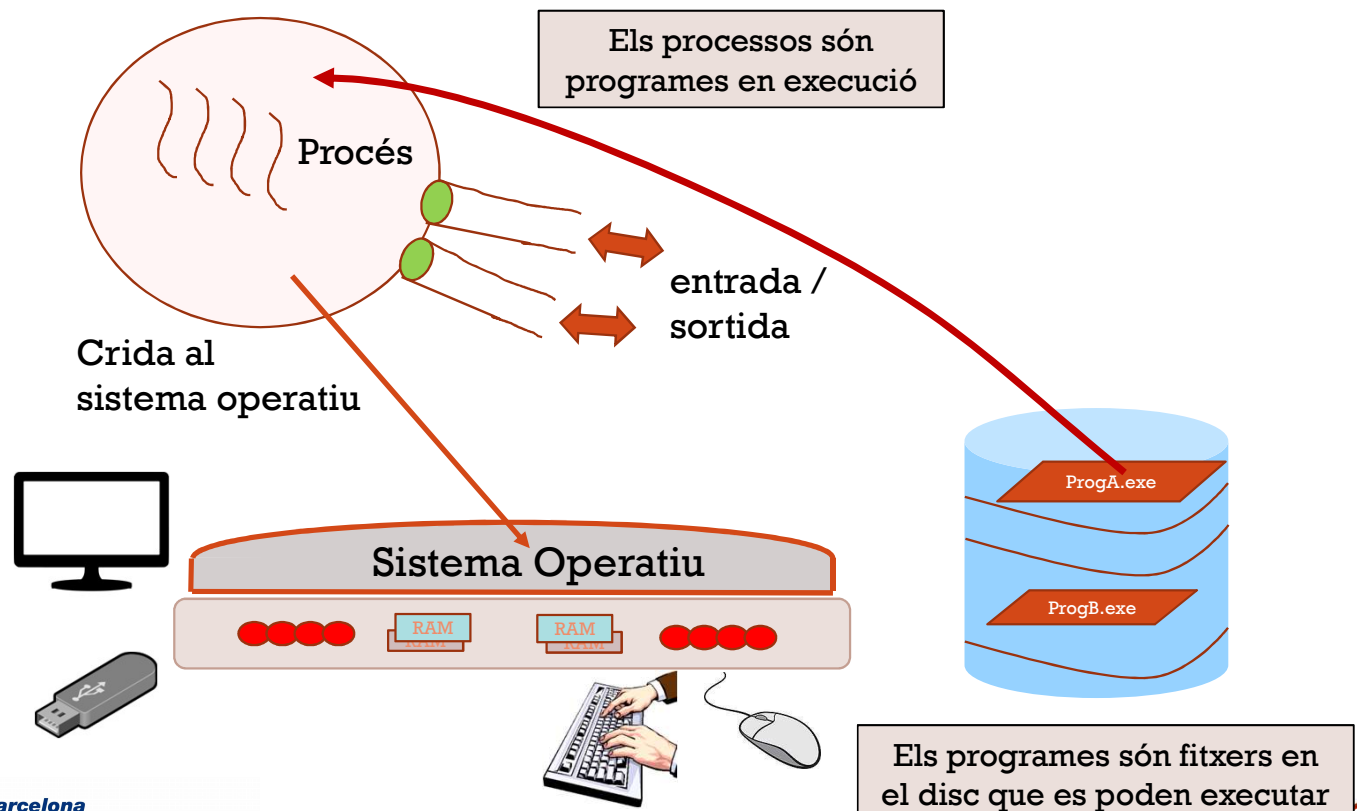
Benvinguda

- Podeu baixar-vos els exemples de:

```
git clone https://github.com/xavim/DeKePuSo.git
```

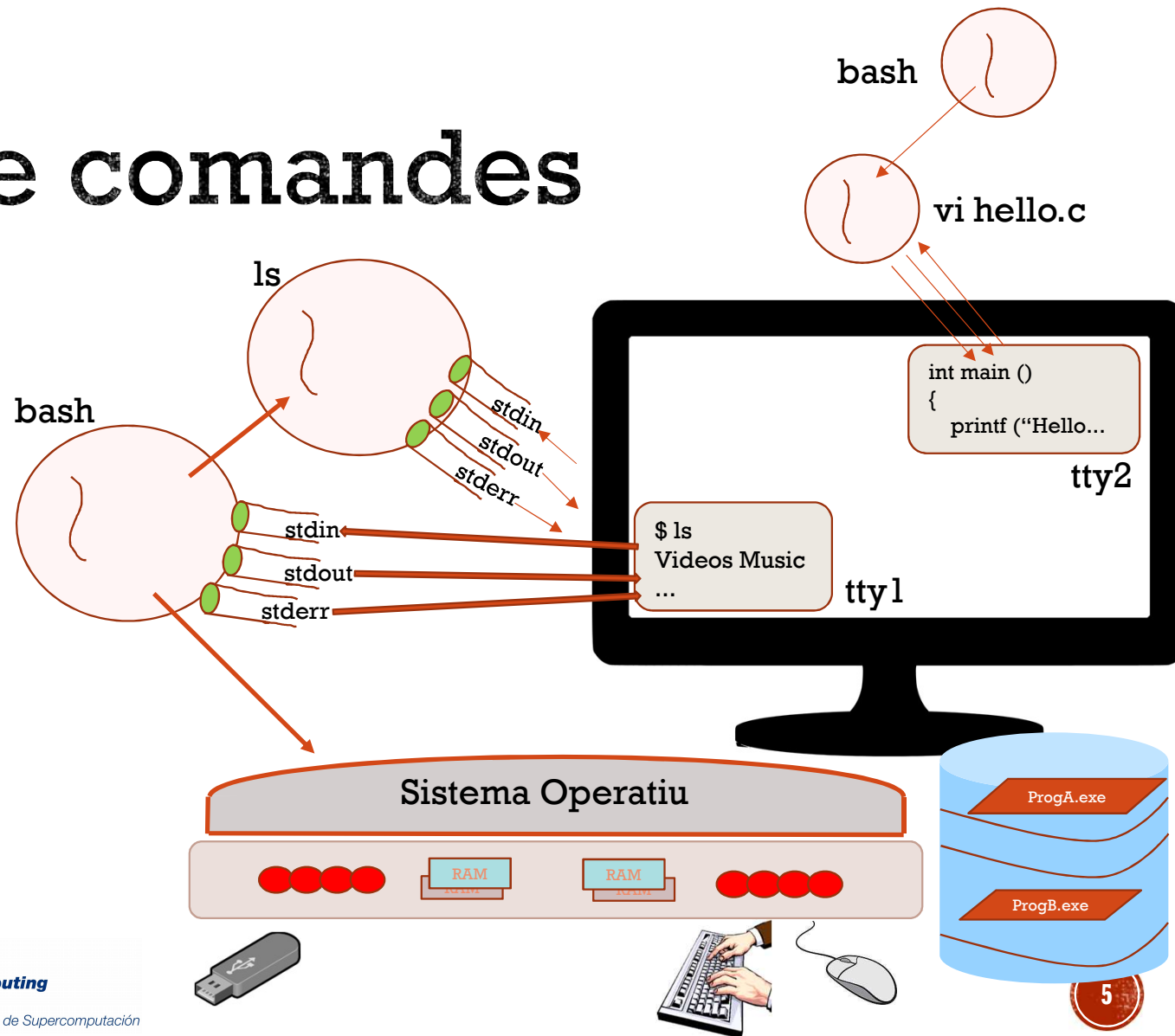
Entorn d'execució

- Sistema operatiu, processos i programes
- Entrada/sortida i dispositius
- Fluxos i paral·lelisme



Intèrpret de comandes

- Un procés més
 - S'encarrega de llegir i interpretar les línies de comandes
 - **bash / sh / csh / tcsh / zsh...**
 - **cmd.exe / powershell.exe...**



Execució de comandes

- La primera paraula de la línia de comandes es considera la comanda per executar

```
$ ls -l
```

- La resta de paraules es consideren els arguments que la comanda rebrà

- Opcions curtes, exemples -l -i -r
- Opcions llargues, exemples ?? --inode --recursive

ATENCIÓ: una mateixa opció té significats diferents per cada comanda

- Podem separar diverses comandes en la mateixa línia amb el ';' com a separador

```
$ hostname ; uptime
```

```
pcxavim6
```

```
17:24:22 up 21:34, 0 users, load average: 0.02, 0.04, 0.08
```

Paraules a la línia de comandes

- Literals `ls mkdir 997 ...`
- Variables (iniciades amb \$) `$PATH $TERM $USER $DISPLAY ${LS_COLORS}`
- Cadenes de caràcters literals `'Hola, Món' '998*997' 'l'USUARI és ${USER}'`
- Cadenes de caràcters amb expansió de variables `"Hola, Món" "998*997" "l'USUARI és ${USER}"`
- Expansió de noms de fitxers `?.c *.c prova[0-9].c prova{2,4,6}.c`
- Redirecció de l'E/S `> out.txt >& out_err.txt < in.txt`

Proveu-les! `$ cd shell`
`$./slide7.sh)`

Variables i expressions

- Podem definir variables amb valors constants

```
$ a=3  
$ echo $a    -> 3
```

- i utilitzar expressions senzilles: fem servir `$((i))`

```
$ echo $((1+3*2)) -> 7  
$ a=$((a+1))  
$ echo $a    -> 4
```

Proveu-les! [./slide8.sh](#)

- Només amb nombres enters (no en punt flotant) i amb els operadors habituals
 - `+ - * / % ++ -- ! ~ ** >> << >= <= < > == != & ^ | && || ?: = op= (== C)`
- Es poden fer servir en condicions de sentències alternatives i bucles `[[$((expressió))]]`

Variables d'entorn

- Les variables “exportades” les hereden els processos fills: són les variables d'entorn

```
$ env                # llista les variables d'entorn i els seus valors  
$ export VARIABLE    # posa VARIABLE a la llista de variables exportades
```

```
CC=gcc  
C_INCLUDE_PATH=/usr/lib/gcc/x86_64-linux/include  
COLORTERM=1  
CPU=x86_64  
CVS_RSH=ssh  
CXX=g++  
F77=f77  
F90=f90  
F95=f95  
FC=gfortran  
HISTFILE=/home/xavim/.bash_history  
HISTSIZE=1000  
HOME=/home/xavim  
HOST=pcxavim6  
HOSTNAME=pcxavim6  
LESS=-M -I -R  
LS_COLORS=no=00:fi=00:di=01;.....  
PATH=/usr/local/bin:/usr/bin:/bin:/usr/sbin:/sbin  
USER=xavim
```

Estructures de control

- Seqüència de comandes -- separades per ‘;’

- `hostname ; uptime`

`pcxavim6`

`17:24:22 up 21:34, 0 users, load average: 0.02, 0.04, 0.08`

- Podem executar comandes usant `...`

- `echo `hostname ; uptime``

- `...` executa les comandes “...” i les substitueix per la seva sortida

`pcxavim6 17:24:22 up 21:34, 0 users, load average: 0.02, 0.04, 0.08`

Estructures de control - if

- Conditionals `if [condició]; then comandes...; else comandes...; fi`

```
if [ "$HOSTNAME" == "pcxavim6" ]; then
    echo som a la màquina 6
else
    echo som a una una màquina no coneguda
fi
```

Proveu-lo! `./slide11.sh`

- Condicions aritmètiques (syntaxi original) `-lt -le -gt -ge (== =)`
- Els caràcters separadors “ ” o TAB (“\t”) són molt importants per tal que aquestes construccions funcionin correctament
- Podeu canviar el nom del host “només per a l’script”:

`HOSTNAME=pcxavim6 ./slide11.sh`

Estructures de control - casos

- Tractament de diferents casos (case, select)

```
case "$HOSTNAME" in
    (pcxavim6) echo a la màquina 6 ;;
    (pcxavim7) echo a la màquina 7 ;;
    (pcxavim8) .... ;;
    ...
    (*) echo a una màquina no coneguda ;;
esac
```

Proveu-lo!!

```
HOSTNAME=pcxavim6 \
./slide12.sh
```

- Els separadors que es poden fer servir:
 - ;; si executa el cas donat, salta directament al final (com el break de C)
 - ;& si executa el cas, continua executant la resta de casos (equiv. a l'absència de break en C)
 - ;;& si executa el cas, continua comprovant la resta de casos (no disponible en C)

Estructures de control - bucles

- Bucle “for” sobre una seqüència d'elements

```
for valor in  valors ...; do  
    comandes ...  
    ...del cos del bucle  
done
```

- Bucle “while” amb una condició arbitrària

```
... inicialització de variables...  
while [ condició ]; do  
    comandes ...  
    ...del cos del bucle  
done
```

Proveu-les!! [./slide13.sh](#)

Sortida al terminal

- Podem controlar la sortida al terminal amb caràcters de control...

- `\a` bell
- `\b` backspace
- `\c` interrupt
- `\e` escape (inici de de seqüència d'escapament)
- `\n` final de línia (form-feed + carriage-return)
- `\r` tornar a inici de línia (carriage-return)
- `\t` tabulador

- i amb seqüències d'escapament (`\x1b`)

Proveu-les!! `./slide14.sh`

<https://www.baeldung.com/linux/bash-escape-characters>

https://pubs.opengroup.org/onlinepubs/9699919799/basedefs/V1_chap06.html

Sortida al terminal

- Algunes seqüències d'escapament...

`\e[A` puja una línia
`\e[B` baixa una línia
`\e[C` avança un caràcter
`\e[D` retrocedeix un caràcter

mou el cursor

`\e[01;30m` lletres en color negre
`\e[01;31m` **vermell**
`\e[01;32m` verd
`\e[01;33m` groc

canvia el color del text

`\e[01;40m` **fons del text en color negre**

canvia el color del fons

...

https://en.wikipedia.org/wiki/ANSI_escape_code

Resum

- Hem vist algunes de les funcionalitats de l'interpret de comandes
 - Comandes i arguments
 - Variables [d'entorn]
 - Expressions
 - Estructures de control
 - Seqüència, condicional / alternativa, bucle
 - Sortida al terminal
- Veieu les pàgines de manual dels diferents interprets
 - bash / csh / tcsh / zsh...
 - Hi ha moltes més característiques interessants!!
 - Control de l'execució (foreground, background), ús de les expressions en condicions amb [[]], taules i taules associatives

Exercici

- Volem saber si una col·lecció d'ordinadors estan funcionant...
 - Farem servir la comanda ping
 - 1 sol ping (-c 1) `ping -c 1 -w 1 <ordinador>`
 - Mínim temps de resposta (-t 1)
 - Tindrem unes taules amb la informació de les màquines
 - `maquines=(www.m1.com www.m2.net www.m3.info www.m4.edu)`
 - `noms=(m1 m2 m3 m4)`
 - Iterarem mentre no premem la tecla 'q' `./slide17.sh --short`
 - Li podrem indicar el format de sortida
 - `[--short | --long]` `./slide17.sh --long`
 - Veieu també els comentaris dins el fitxer de l'script

Proveu de fer-lo vosaltres!
I després compareu-lo amb la nostra solució