

SISTEME DE OPERARE (SO)

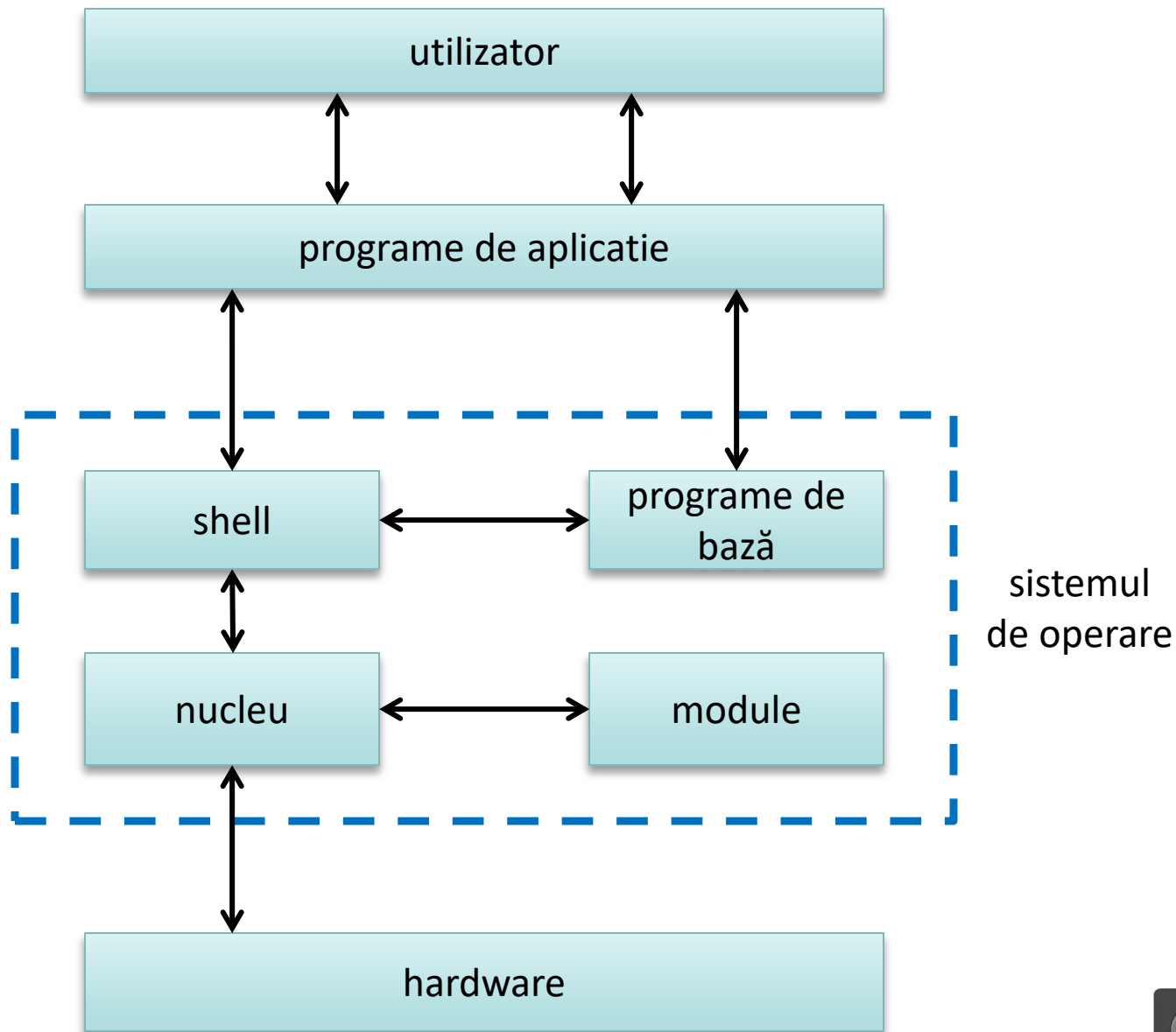
CURS 5

*Lect. Univ. Dr. Mihai Stancu*

INTERFATA IN LINIA DE COMANDA

- Suport (Introducere în sisteme de operare)
  - Capitolul 12 – Shell scripting
    - Secțiunile 12.1, 12.2, 12.6, 12.7

# STRUCTURA UNUI SISTEM DE CALCUL



- interacțiunea utilizatorului cu aplicațiile și cu sistemul de operare
- ascunde detaliile sistemului de operare
- ușurința în utilizare (*ease of use*)
- eficiența în utilizare
- acomodarea rapidă: intuitivă, naturală
- interacțiunea cu sistemul de calcul
- *Human Machine Interface*

- GUI – Graphical User Interface
  - WIMP (window, icon, menu, pointing device)
  - eye-candy
  - ușor de folosit, intuitivă
  - ascunderea complexității
- CLI – Command Line Interface
  - execuția de comenzi pentru îndeplinirea sarcinilor
  - eficiență
  - automatizarea sarcinilor
  - resurse consumate reduse

```
zlai@ZLai: ~  
zlai@ZLai:~$ uname  
Linux  
zlai@ZLai:~$ uname -a  
Linux ZLai 3.2.0-24-generic #39-Ubuntu SMP Mon May 21 16:52:17 UTC 2012 x86_64 x  
86_64 x86_64 GNU/Linux  
zlai@ZLai:~$ which which  
/usr/bin/which  
zlai@ZLai:~$ which yes  
/usr/bin/yes  
zlai@ZLai:~$ echo $SHELL  
/bin/bash  
zlai@ZLai:~$ echo $PATH  
/usr/lib/lightdm/lightdm:/usr/local/sbin:/usr/local/bin:/usr/sbin:/usr/bin:/sbin  
:/bin:/usr/games  
zlai@ZLai:~$ cat ./yes  
#!/bin/bash  
#script version of /usr/bin/yes  
  
while true; do  
    echo y  
done  
zlai@ZLai:~$ echo $SHELL  
/bin/bash  
zlai@ZLai:~$ echo $?  
0  
zlai@ZLai:~$  
zlai@ZLai:~$  
zlai@ZLai:~$ pwd  
/home/zlai  
zlai@ZLai:~$
```

# CLI

```
PS C:\> Get-ChildItem 'MediaCenter:\Music' -rec |
>> where < -not $_.PSIsContainer -and $_.Extension -match 'wma|mp3' > |
>> Measure-Object -property length -sum -min -max -ave
>>

Count       : 1307
Average     : 5491276.09563887
Sum         : 7177097857
Maximum     : 22905267
Minimum     : 3235
Property    : Length

PS C:\> Get-WmiObject CIM_BIOSElement | select biosv*, man*, ser* | Format-List

BIOSVersion : <TOSCP - 6040000, Ver 1.00PARTTBL>
Manufacturer : TOSHIBA
SerialNumber : M821116H

PS C:\> <[wmiSearcher]@'
>> SELECT * FROM CIM_Job
>> WHERE Priority > 1
>> '@.get() | Format-Custom
>>

class ManagementObject#root\cimv2\Win32_PrintJob
{
    Document = Monad Manifesto - Public
    JobId = 6
    JobStatus =
    Owner = User
    Priority = 42
    Size = 1027088
    Name = Epson Stylus COLOR 740 ESC/P 2, 6
}

PS C:\> $rssUrl = 'http://blogs.msdn.com/powershell/rss.aspx'
PS C:\> $blog = [xml](new-object System.Net.Webclient).DownloadString($rssUrl)
PS C:\> $blog.rss.channel.item | select title -first 3

title
-----
MMS: What's Coming In PowerShell U2
PowerShell Presence at MMS
MMS Talk: System Center Foundation Technologies

PS C:\> $host.version.ToString().Insert(0, 'Windows PowerShell: ')
Windows PowerShell: 1.0.0.0
PS C:\>
```

- prompt + comenzi
- tipuri de utilizatori
  - programatori
  - administratori de sistem/rețea
  - utilizatori avansați din punct de vedere tehnic
- shell-ul este un caz particular de CLI
- alte exemple de CLI
  - consola GRUB (bootloader)
  - consola Python
  - consola de configurare de echipamente de rețea
  - consola Matlab, AutoCAD
  - consolă de jocuri (FPS)



## SHELL

- aplicație care permite accesul la resursele sistemului de operare
- poate fi grafic (GUI) sau în linia de comandă (CLI)
- în Windows shell-ul este Explorer (GUI)
- în Unix
  - shell GUI: X Window System + KDE/GNOME/Xfce
  - shell-uri CLI: bash, tcsh, zsh, dash
- Cum putem rula o sesiune de shell în Linux?
  - terminal virtual (ALT+CTRL+F1...F6)
  - emulator de terminal (konsole, gnome-terminal, xterm etc.)
  - conexiune de la distanță
- terminalul este (pseudo)dispozitivul în care rulează shell-ul
- în continuare, shell = shell CLI

- terminal = dispozitiv
- shell = proces
- terminal – cadru, interfață pentru rularea unui shell
  - fizic (tastatură, monitor)
  - virtual (ALT+CTRL+F1. . . F6) – tty0, tty1, . . . , tty6
  - emulator de terminal – pts/0, pts/1, . . .
  - folosit pentru intrarea/ieșirea standard (stdin/stdout) a unui proces
  - daemonii nu au acces la nici un terminal (nu au intrare/ieșire standard)
- shell – permite rularea de comenzi
  - prompt, comenzi, operatori shell
  - bash, zsh, tcsh

Promptul unui shell

```
alin@anaconda:~$ whoami  
alin
```

- comenzile pot avea argumente (opțiuni pentru comandă)
  - argumentele se separă prin spații
- Keep it simple!
  - nume simple, uneori neintuitive: cp, rm, ls, vi, ps
- Cum te documentezi legat de o comandă?
  - --help
  - comenzile man/info
  - Google (“tutorial”, “faq”, “how to”, “example”, “usage”)

## FUNCTIONAREA UNUI SHELL

- se introduce o comandă (`ls`)
- shell-ul identifică executabilul asociat (`/bin/ls`)
- este creat un nou process shell (`/bin/bash`) (`fork`)
- procesul creat se “transformă” conform cu nou executabil
- intrarea standard este controlată de noul proces
- după execuția comenzii, procesul moare și controlul revine shell-ului

## FACILITATI SHELL

Afișare colorată

```
ls --color=auto
```

Personalizare prompt

```
alin@anaconda:~/junk/uso/curs8/tmp$ echo $PS1  
$debian_chroot:+($debian_chroot)\u@\h:\w\$_
```

- command completion
  - folosirea tastei TAB
  - eficiență
  - completarea comenzii și a argumentelor acesteia
  - `/etc/bash_completion`
- biblioteca readline (`libreadline`)
  - keyboard shortcuts (eficiență în utilizare)
  - documentare: `man bash` și căutare după *Commands for*

- operatori
- redirectare: >, <, &>, >>, <<
- secvențiere, înlănțuire: ;, ||, &&, |, &
- expandare: \$
- comentare: #
- citare (escaping): ', ", \
- separare: blank (spațiu)
- globbing: ?, \*, [, ], {, }

Redirectarea intrării/ieșirii: >, <

```
alin@anaconda:~/junk/uso/curs8/tmp$ ls -l > out.txt
alin@anaconda:~/junk/uso/curs8/tmp$ wc -l < out.txt
4
```

Comunicația între procese: |

```
alin@anaconda:~/junk/uso/curs8/tmp$ ls -l | wc -l
4
```

Rularea unui proces în background: &

```
alin@anaconda:~$ du -hs &
[1] 15766
alin@anaconda:~$ jobs
[1]+  Running                  du -hs &
```

## CE EFECT ARE COMANDA DE MAI JOS?

```
me@local:~$ < /dev/urandom tr -d -c 'A-Za-z0-9' | head -c 8 ; echo
```

- extrage primele 8 caractere tipăribile (literă sau cifră) dintr-un șir generat aleator; la final afișează un caracter *newline*
  - șterge orice nu este literă sau cifră dintr-un șir infinit generat aleator
    - `/dev/urandom` – generator de caractere aleatoare
    - `< /dev/urandom` – redictează intrarea standard din fișierul `/dev/urandom`
    - `tr` – *transliterate* (comandă shell de tip **filtru de text**)
    - `-d` – *delete* (șterge)
    - `-c` – *complement* (set complementar)
    - `-c 'A-Za-z0-9'` – orice nu este literă sau cifră
  - extrage primele 8 caractere
    - `head` – extrage primele elemente
    - `-c 8` – opt caractere



### Inițializare variabile în shell

```
alin@anaconda:~$ a=10
alin@anaconda:~$ b=ana
alin@anaconda:~$ c="ana are mere"
alin@anaconda:~$ d="$b $a"
alin@anaconda:~$ echo $a, $b, $c, $d
10, ana, ana are mere, ana 10
```

- **nu** se folosește spațiu înainte și după egal la inițializare
  - s-ar considera comandă cu argumente
- folosirea valorii unei variabile se realizează prin prefixarea cu simbolul \$ (*dollar*)
- variabilele shell **nu** au tip (întreg, șir, pointer)

- documentație: `man bash`, și căutare după *Special*

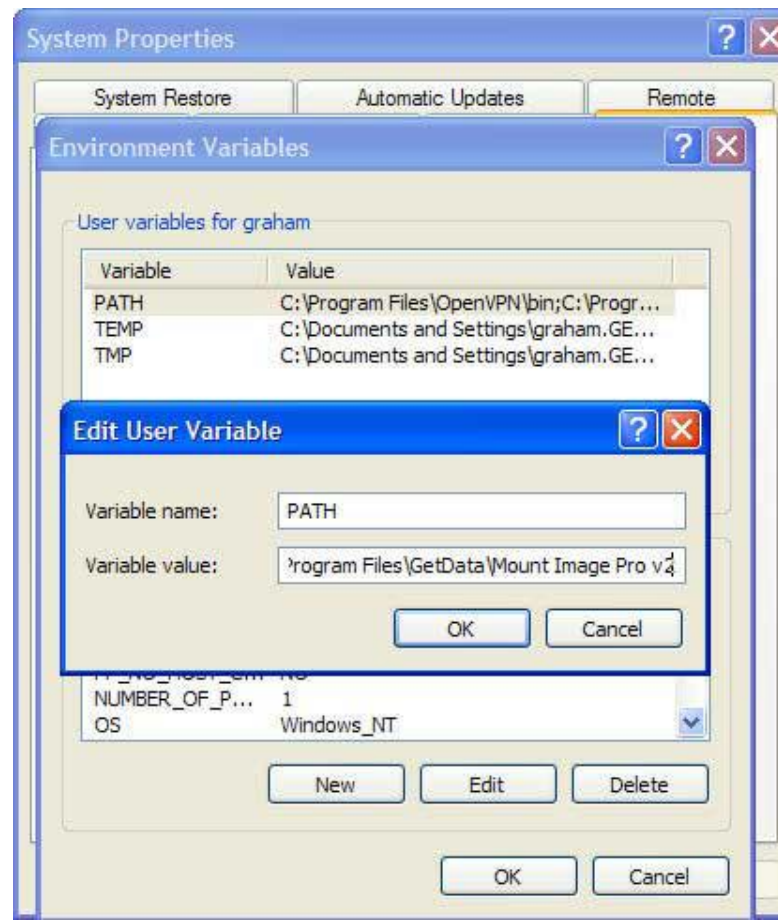
### *Parameters*

- `$?` – valoarea de retur a ultimei comenzi
- `$$` – PID-ul procesului shell curent
- `$!` – PID-ul ultimului proces (job) lansat în background
- `$_` – ultimul argument al ultimei comenzi

- definesc contextul de rulare a unui proces
- sunt moștenite de procesele copil
- exemple: PATH, HOME, PWD, USERNAME, SHELL
- operații specifice cu variabile de mediu
  - listare: `env`, `printenv`
  - configurare variabilă ca variabilă de mediu (exportare)
    - `export myvar`
    - `export myvar=42 # init and export`
  - persistența configurării (*startup configuration*)
    - în fișierul `~/ .bashrc`

## ➤ în Windows

- **per sistem:** Control Panel -> System -> Advanced -> Environment Variables
- **per utilizator:** Control Panel -> Performance and Maintenance -> System



### Exemple de escaping

```
alin@anaconda:/tmp/uso$ ls -l
total 0
-rw-r--r-- 1 alin alin 0 2007-11-18 00:14 alfa beta.txt
-rw-r--r-- 1 alin alin 0 2007-11-18 00:14 d.txt
alin@anaconda:/tmp/uso$ ls alfa beta.txt
ls: alfa: No such file or directory
ls: beta.txt: No such file or directory
alin@anaconda:/tmp/uso$ ls "alfa beta.txt"
alfa beta.txt
alin@anaconda:/tmp/uso$ ls 'alfa beta.txt'
alfa beta.txt
alin@anaconda:/tmp/uso$ ls alfa\ beta.txt
alfa beta.txt
alin@anaconda:/tmp/uso$ echo $a
4
alin@anaconda:/tmp/uso$ echo "$a"
4
alin@anaconda:/tmp/uso$ echo \$a
$a
alin@anaconda:/tmp/uso$ echo '$a'
$a
```

*CITARE (ESCAPING) – CONTINUARE*

- escaping folosind \ (backslash)
  - forțează forma literală a caracterului următor
  - rol special în cazul comenzii `echo -e`

```
alin@anaconda:~$ echo -e "a\tb\tc"
a      b      c
```

- escaping folosind " " (ghilimele, quote)
  - escapează textul dintre ghilimele
  - nu escapează expandarea

```
alin@anaconda:~$ echo "* ; && | > $sir"
* ; && | > file.txt
```

- ' (apostrof, single quote)
  - escapează orice între apostroafe

```
alin@anaconda:~$ echo '\\\$a$bc$de'
\\\$a$bc$de
```

## EXPANDAREA IN SHELL

### Expandarea variabilelor

```
alin@anaconda:~$ a=3
alin@anaconda:~$ echo $a
3
```

### Expandarea comenzilor

```
alin@anaconda:~$ ls | wc -l
17
alin@anaconda:~$ num=$(ls | wc -l)
alin@anaconda:~$ echo $num
17
```

### Expandarea aritmetică

```
alin@anaconda:~$ a=3
alin@anaconda:~$ ((a++))
alin@anaconda:~$ b=$((a+1))
alin@anaconda:~$ echo $b
5
```

## Folosire globbing în comenzi shell

```
alin@anaconda:~/code/tests$ ls
Makefile endian.c struct_bit struct_bit.o tcp_new_sock.c
endian endian.o struct_bit.c tcp_new_sock tcp_new_sock.o
alin@anaconda:~/code/tests$ ls *.c
endian.c struct_bit.c tcp_new_sock.c
alin@anaconda:~/code/tests$ ls end?an.c
endian.c
alin@anaconda:~/code/tests$ ls [a-s]*.c
endian.c
alin@anaconda:~/code/tests$ ls *.c,o
endian.c endian.o struct_bit.c struct_bit.o
tcp_new_sock.c tcp_new_sock.o
```



- filename expansion (și altele)
- diferit de expresii regulate (vezi în continuare)
- \* – orice caracter de oricâte ori
- ? – orice caracter
- [abcde], [a-e] – set de caractere
- [^abc] – diferit de setul de caractere
- {jpg,png,bmp} – selecție între variante

*EXPRESII REGULATE*

- **grep** - global regular expression print

Extrage liniile care încep cu șirul `ana`

```
alin@anaconda:/tmp/for$ grep '^ana' /etc/passwd
ana:x:1118:1026:Ana Musat:/home/students/ana:/bin/bash
```

Extrage liniile care conțin cu șirul `ana`

```
alin@anaconda:/tmp/for$ grep 'ana' /etc/passwd
list:x:38:38:Mailing List Manager:/var/list:/bin/sh
roxana:x:1004:1004:Roxana
Savu,,,:/home/users/roxana:/bin/bash
roxanam:x:1030:1026:Roxana
Moraru:/home/students/roxanam:/bin/bash
[...]
```

Listează fișierele ce conțin linii care conțin doar șirul `#include <stdio.h>`

```
alin@anaconda:/tmp/for$ grep -r '^#include <stdio\.h>$'
/usr/include
/usr/include/argp.h:#include <stdio.h>
/usr/include/mntent.h:#include <stdio.h>
/usr/include/printf.h:#include <stdio.h>
```

- **diferite** de globbing
- $x^*$  – caracterul  $x$  de oricâte ori (posibil niciodată)
- $x^+$  – caracterul  $x$  de oricâte ori (cel puțin o dată)
- $x?$  – caracterul  $x$  cel mult o dată
- $.$  (punct) – orice caracter
- $^$  – început de rând
- $\$$  – sfârșit de rând
- $[a-z]$  – set de caractere
- $[^a-z]$  – set complementar de caractere
- $a|b$  – caracterul  $a$  sau  $b$
- **Atenție:** la grep unele trebuie escapate ( $\backslash+$ ,  $\backslash|$ )

- 1) expresie regulată pentru adrese MAC (0C:A3:11:CF:D1)
- 2) expresie regulată pentru subgrupă de student la Informatică (212A, 223B)
- 3) expresie regulată pentru nume de variabile C
  - încep cu litere sau cu underscore
  - pot conține litere, underscore sau cifre
- 4) expresie regulată pentru nume valide Windows
  - nu încep cu spațiu sau punct
  - nu conțin \, /, \*, ?, :, ", <, >, |

- ediția a 3-a, 2002
- enciclopedie a facilităților Unix
- 1100 de pagini, 51 de capitole
- pentru „power users” – utilizatori avansați care vor să folosească la maxim sistemul
- capitole excelente despre shell și shell scripting

- parte a echipei Unix
- ulterior, dezvoltator la Plan 9 și Inferno
  - sisteme de operare unde totul este un fișer
  - pe Unix aproape totul este un fișier
- cărți și proiecte împreună cu Ken Thompson, Brian Kernighan
- UTF-8
- lucrează la Google



- Amazon.com, Inc.
- comerț electronic (online retailer)
- fondată în 1994, online în 1995
- inițial librărie online
- Amazon Kindle
- în 2010, vânzările de e-books au depășit vânzările de cărți în format printat
- Amazon Web Services – servicii de stocare și rulare, cloud computing
  - Amazon EC2 – închiriere de mașini virtuale
  - Amazon S3 – servicii de stocare

- CLI pentru Windows
- rulează de la Windows XP SP2 în sus
- integrare cu .NET framework
- *cmdlets*, aplicații .NET
- automatizare – scripting
- pipeline de obiecte (*objects*), nu de șiruri de caractere ca în Unix



## CUVINTE CHEIE

- interfață
- CLI
- shell
- terminal
- comandă, argumente
- prompt
- >, <, &, |, ||, &&, ;
- command completion
- ", ', \, \$
- variabile
- variabile de mediu
- escaping
- expandare
- globbing
- expresii regulate
- grep

- [http://en.wikipedia.org/wiki/User\\_interface](http://en.wikipedia.org/wiki/User_interface)
- <http://www.gnu.org/software/bash/manual/bashref.html>
- <http://mywiki.woledge.org/BashFAQ>
- <http://www.shelldorado.com/>
- <http://www.pixelbeat.org/cmdline.html>
- <http://www.regular-expressions.info/>