

BASH Scripting

Luís Ferreira André Santos Pedro Faria

Universidade do Minho
CAOS Talks

`zamith@groupbuddies.com`
`andrefs@groupbuddies.com`
`pedro.faria.80@gmail.com`

31 de Maio de 2011

- Operador >

Operador >

```
$ echo Hello > testFile
```

```
$ cat testFile
```

```
Hello
```

Pipes e redireccionamento

- Operador >
- Operador >>

Operador >>

```
$ echo Hello2 » testFile
```

```
$ cat testFile
```

```
Hello
```

```
Hello2
```

Pipes e redireccionamento

- Operador >
- Operador >>
- Operador <

Operador <

```
$ cat < testFile
```

```
Hello
```

```
Hello2
```

Pipes e redireccionamento

- Operador >
- Operador >>
- Operador <
- Pipe |

Pipe |

```
$ cat testFile | sort -r  
Hello2  
Hello
```

Pipes e redireccionamento

- Operador >
- Operador >>
- Operador <
- Pipe |
- Redireccionar file descriptors

Redireccionar file descriptors

```
$ rm direct/  
rm: direct/: is a directory  
$ rm direct/ 2> error  
$ rm direct/ 2>&1  
rm: direct/: is a directory
```

Pipes e redireccionamento

- Operador >
- Operador >>
- Operador <
- Pipe |
- Redireccionar file descriptors
- Auto-redireccionamento
inline

HERE Documents

```
$ bc << HERE  
> 12+12 -5  
> 100*10-200  
> HERE  
19  
800
```

5 coisas que deves saber antes de escrever um script

❶ Há 3 tipos de quotes:

Aspas Aceitas espaços e expande variáveis

Pelicas Aceita espaços mas não expande variáveis

Backslash Faz escape ao valor de .

❷ *Globbering*

- *
- {...}
- outros (?, [...], [^...])

❸ \$(comando), encapsula o *output* do comando

❹ var="ls ."¹

❺ Comentários são feitos com o #

¹Não pode haver espaços entre o igual, e se houver espaços no valor este tem de estar entre aspas

O primeiro script

```
1 #!/bin/bash
2
3 echo "Name of file to cat:"
4 read file
5 cat $file
6
7 greeting="\nHello World"
8 echo -e $greeting
9
10 exit 0
```

- A linha 1 diz à shell qual o interpretador a usar e começa sempre com `#!`
- Por norma um script que termine correctamente deve retornar o valor 0

Para correr o script basta dar permissões de execução a toda a gente (ou não), com `chmod +x file.sh` e corrê-lo `./file.sh`

Comando do minuto

Variáveis de ambiente

No início de qualquer script bash temos acesso a algumas variáveis, que podem ser alteradas com um export ou no .bashrc

\$HOME Home do utilizador actual

\$PATH Lista de directorias onde são procurados os comandos

\$PS1 Definição da prompt (ex: \h:\W \u \s)

\$PS2 Prompt secundária, normalmente >

\$IFS *Input Field Separator*. Lista de caracteres usados para separar palavras quando a ler do *input*

\$0 O nome do script

\$# O número de parâmetros passados

\$\$ O PID do script (normalmente usado para criar ficheiros temporários)

Para ver todas as variáveis de ambiente, *printenv*²

²http://tldp.org/LDP/Bash-Beginners-Guide/html/sect_03_02.html

Variáveis dos parâmetros

Este tipo de variáveis são um subconjunto das de ambiente, que dizem respeito aos parâmetros recebidos pelo script

- \$1, \$2, ... Os parâmetros passados, por ordem
- \$* Lista de todos os parâmetros, separadas pelo primeiro caracter em IFS
- @ Uma variação de \$*, que usa sempre o espaço como separador (aconselhado)