

Operating Systems

Dmitry Zaitsev

Lecture 5:

**Case study: Pipes and filters.
Programming in shell language.**

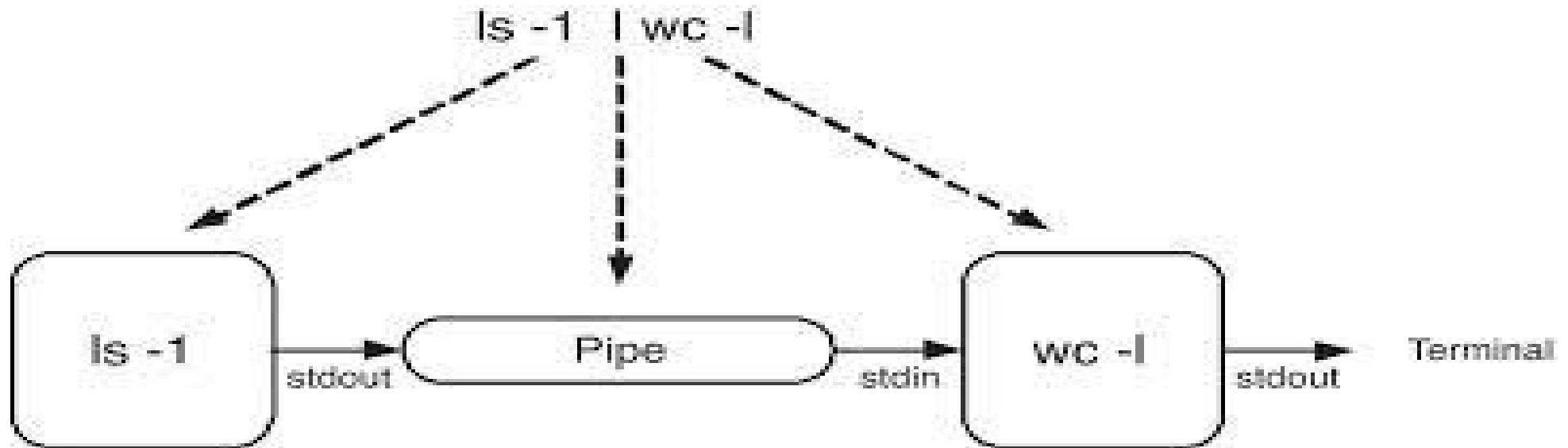
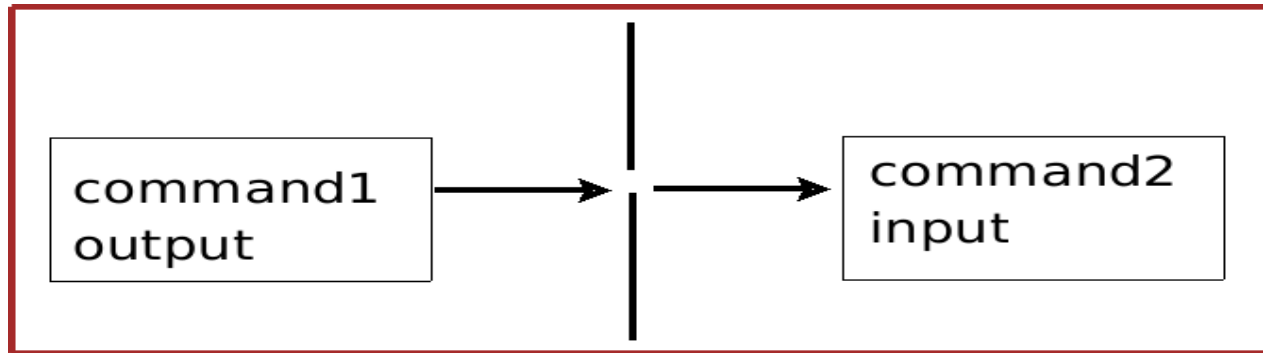
Redirecting input & output in Linux



- `cat > file`
- `ls -l >> file`
- `wc < file`
- `ls qq 2> errors`
- `ls qq 2>&1`



Pipes



Filters

Utilities designed to work mostly within pipes

- `ls | more`
- `uniq, cut, fmt, head, tail, tee, wc`
- `sort, grep, tr, sed, awk`

- `ps -el | head`
- `ps -el | cut -b 5-9 | uniq -u`

Sort

- `sort` - sort lines of text files
- `-n` numerical, `-r` reverse, `-m` merge, `-u` unique
- `-k` sort via key: `F[.C][OPTS][, F[.C][OPTS]]`
- `cat /etc/passwd | sort`
- `ls -l | sort -k 9r`
- `ps -e | sort -k 2`

daze@daze-fast: ~

```
drwxrwxr-x 4 daze daze 4096 anp. 28 15:59 dima
drwxr-xr-x 2 daze daze 4096 anp. 27 11:27 Documents
drwxr-xr-x 2 daze daze 4096 anp. 27 11:27 Downloads
-rw-r--r-- 1 daze daze 8980 anp. 27 10:29 examples.desktop
drwxr-xr-x 2 daze daze 4096 anp. 27 11:27 Music
drwxr-xr-x 2 daze daze 4096 anp. 27 11:27 Pictures
drwxr-xr-x 2 daze daze 4096 anp. 27 11:27 Public
drwxr-xr-x 2 daze daze 4096 anp. 27 11:27 Templates
drwxr-xr-x 2 daze daze 4096 anp. 27 11:27 Videos
```

daze@daze-fast:~\$ ls -l|sort -k 9r

```
drwxr-xr-x 2 daze daze 4096 anp. 27 11:27 Videos
drwxr-xr-x 2 daze daze 4096 anp. 27 11:27 Templates
drwxr-xr-x 2 daze daze 4096 anp. 27 11:27 Public
drwxr-xr-x 2 daze daze 4096 anp. 27 11:27 Pictures
drwxr-xr-x 2 daze daze 4096 anp. 27 11:27 Music
-rw-r--r-- 1 daze daze 8980 anp. 27 10:29 examples.desktop
drwxr-xr-x 2 daze daze 4096 anp. 27 11:27 Downloads
drwxr-xr-x 2 daze daze 4096 anp. 27 11:27 Documents
drwxrwxr-x 4 daze daze 4096 anp. 28 15:59 dima
drwxr-xr-x 2 daze daze 4096 anp. 27 11:27 Desktop
-rw-rw-r-- 1 daze daze 16 anp. 28 12:04 c
```

total 52

daze@daze-fast:~\$

Grep

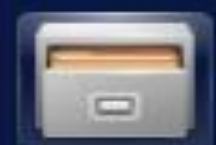
- grep - print lines matching a pattern
 - -n line number, -r recursive, -i ignore case
 - regular expression: ?, *, +, [list], [^list], [range]
 - anchor: “^” beginning, “\$” end
-
- ls -l | grep -i music
 - ls -l | grep s\$



En



09:28



daze@daze-fast: ~

daze@daze-fast:~\$

daze@daze-fast:~\$ ls -l

total 52

-rw-rw-r--	1	daze	daze	16	anp.	28	12:04	c
drwxr-xr-x	2	daze	daze	4096	anp.	27	11:27	Desktop
drwxrwxr-x	4	daze	daze	4096	anp.	28	15:59	dima
drwxr-xr-x	2	daze	daze	4096	anp.	27	11:27	Documents
drwxr-xr-x	2	daze	daze	4096	anp.	27	11:27	Downloads
-rw-r--r--	1	daze	daze	8980	anp.	27	10:29	examples.desktop
drwxr-xr-x	2	daze	daze	4096	anp.	27	11:27	Music
drwxr-xr-x	2	daze	daze	4096	anp.	27	11:27	Pictures
drwxr-xr-x	2	daze	daze	4096	anp.	27	11:27	Public
drwxr-xr-x	2	daze	daze	4096	anp.	27	11:27	Templates
drwxr-xr-x	2	daze	daze	4096	anp.	27	11:27	Videos

daze@daze-fast:~\$ ls -l | grep -i music

drwxr-xr-x 2 daze daze 4096 anp. 27 11:27 Music

daze@daze-fast:~\$ ls -l | grep s\$

drwxr-xr-x	2	daze	daze	4096	anp.	27	11:27	Documents
drwxr-xr-x	2	daze	daze	4096	anp.	27	11:27	Downloads
drwxr-xr-x	2	daze	daze	4096	anp.	27	11:27	Pictures
drwxr-xr-x	2	daze	daze	4096	anp.	27	11:27	Templates
drwxr-xr-x	2	daze	daze	4096	anp.	27	11:27	Videos

daze@daze-fast:~\$

Bash

- *Bourne again shell*
- *Arguments*
- *Built-in commands: echo, pwd, read*
- *History*
- *Expansions: ~, \$, *, ?, ;, ", `*
- *Variables: A=5*
- *Environment variables: \$HOME, \$PATH*
- *Sequence control: if, while, until, for, case, select*
- *Execute a script – commands from a file*

Expansion

- Brace expansion: `a{d,c,b}e`
- Tilde expansion: `"~"` - login name
- Parameter expansion: `"$"` - parameters, command, arithmetic expression;
`${parameter}`, `$(command)`, `$((expression))`
- Pathname expansion: `*`, `?`, `[...]`
- Here document: `<<<word ... word`

Variables, arrays, and expressions

- Variable = Value
- Variable[Index] = Value
- A="Hello VU"
- echo \$A

- X=5
- Y=6
- expr \$X + \$Y

Command substitution

- `$(command)`
- ``command``
- `A=$(ls -l | grep di)`
- `echo prefix${A}suffix`

Branching – if

- if list ; then list ; [elif list ; then list ;] ... [else list ;] fi
- if list; then list; else list; fi
- if list; then list; fi

- if /bin/true; then echo true is true; fi
- if ls; then echo listed; fi

Evaluation of expressions

- `[[expression]]`
 - test expression
 - `>`, `<`, `>=`, `<=`, `==`, `!=`
 - Files: `-a`, `-d`, `-f`, `-x`
-
- `if [[5 < 3]] ; then echo a; else echo b; fi`
 - `if [-f abc]; then echo there is abc; fi`
 - `if test -f abc ; then echo there is abc; fi`

Loop – while

- while list ; do list ; done

```
A=1 S=0
```

```
while [[ $A < 5 ]]; do
```

```
S=$(( $S + $A ))
```

```
A=$(( $A + 1 ))
```

```
done echo $S
```

Loop – for

- for name [[in [word ...]];] do list ; done
- for i in 1 2 3 ; do echo \$i ; done
- for n in `ls` ; do echo name is \$n ; done

Shell functions and export

- function name [()] compound-command
- export Variable

- function pl () {ls; ps}
- export pl
- pl

Shell scripts

```
cat > dummy
```

```
#!/bin/bash
```

```
while /bin/true; do $1; done Ctrl^C
```

```
chmod +x dummy
```

```
./dummy "I'm dummy"
```


Parameters

- \$1 \$2 ...
- \$*, @\$ - all parameters
- \$# - number of parameters
- \$? - exit status
- \$\$ - shell PID

Shell variables

- BASH, BASHPID, BASH_VERSION
 - HOSTNAME, PPID, EFS
 - HOME, PATH
 - PS1, PS2, ...
-
- `cd $HOME`
 - `echo $PATH`

Command line prompt format

- \d - date
- \h - hostname
- \u - username
- \t - time
- PS1="\h:\u>"
- daze-fast:daze>

man bash