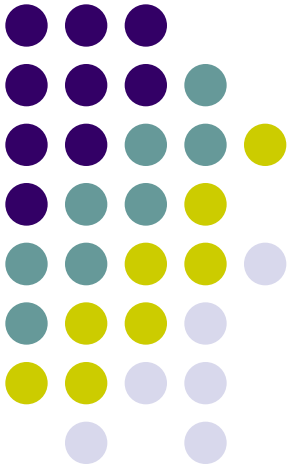
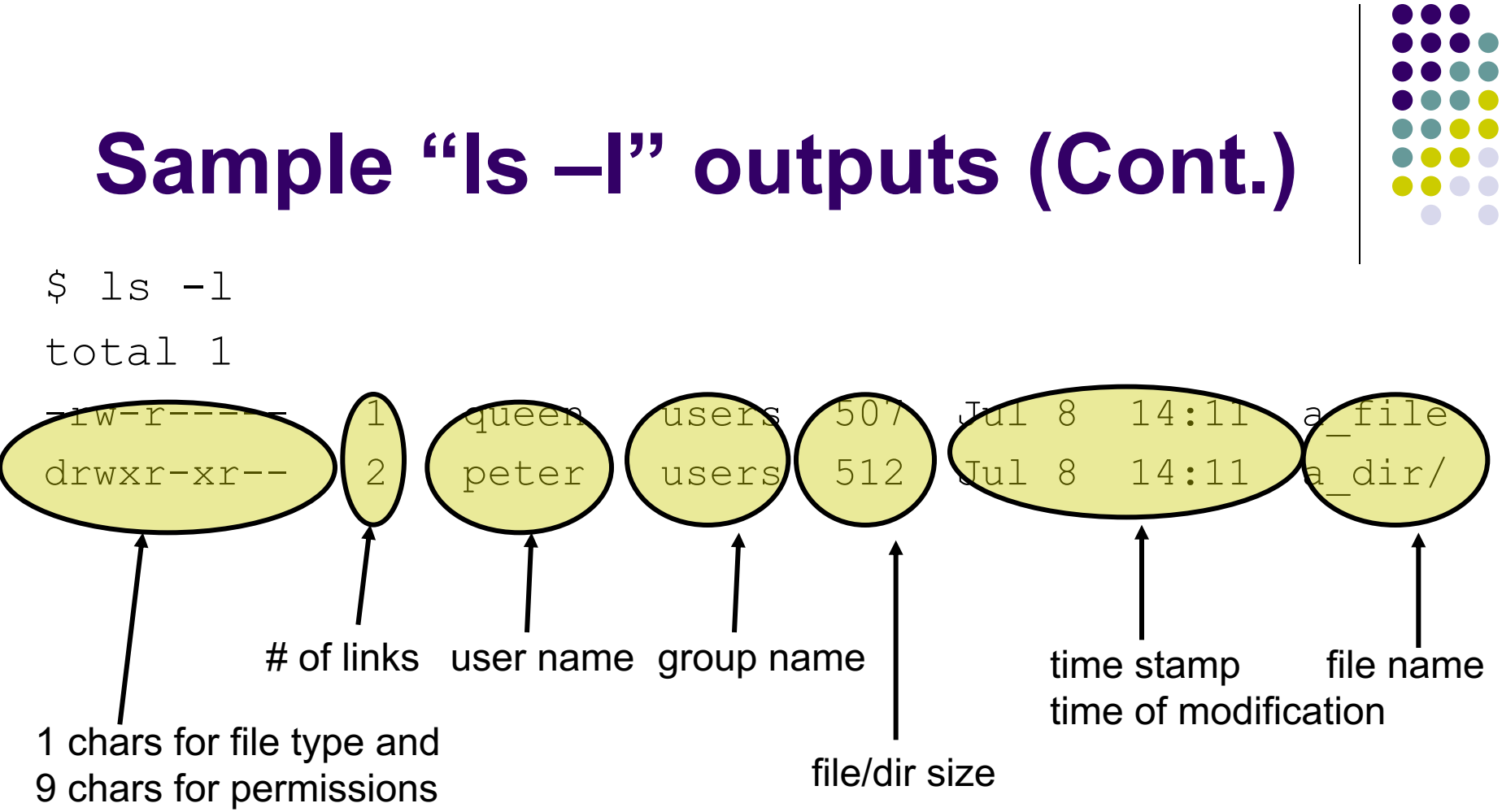
Introduction to Unix Commands

Unix指令簡介

劉靖家

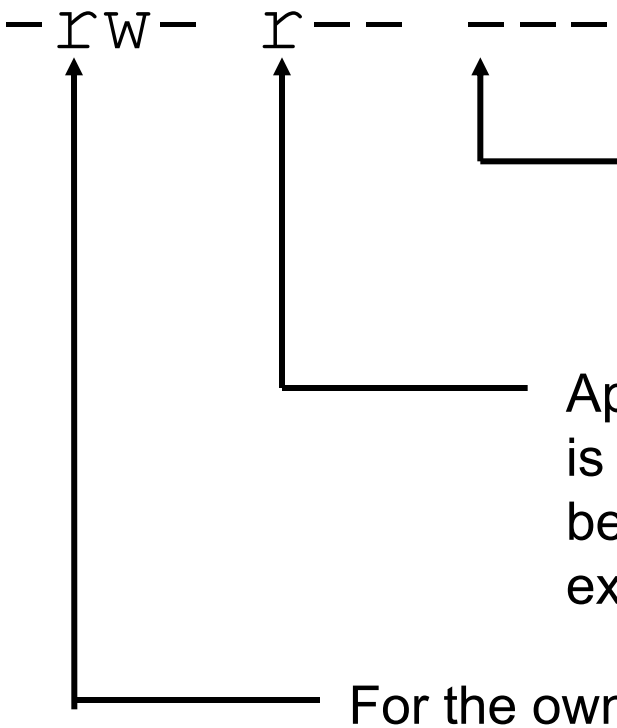


Sample “ls –l” outputs (Cont.)



FYI, all these information are stored on disk with a data structure called “inode”.

File Permission Exmples

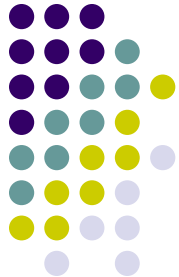


Apply to any user who is not queen and is not a member of the users group. Those users won't have any rights on the file at all.

Apply to any user who is not queen but who is a member of the users group. They will be able to read the file (r), but not write or execute it (--)).

For the owner, queen, has the right to read the file (r), to modify its contents (w) but not to execute it (-).

Directory Permission Exmples



```
drwx r-x r--
```

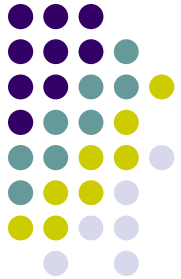
Every other user will only be able to list the contents of this directory (r). Because they don't have wx permissions, they won't be able to write files or enter the directory.

Each user who is not peter, but a member of the users group, will be able to list files in this directory (r), but not remove or add files (-), and will be able to traverse it (x)

peter, as the directory owner, can list files contained inside (r), add to or remove files from that directory (w), and can traverse it (x)

Note: "root" user has the capability to change every permission of any files in the system.

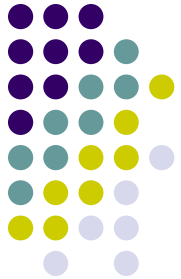
pwd (Print Working Directory)



- Print the absolute pathname of the current working directory
- Example

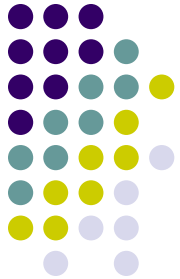
```
$ pwd
/home/queen
```

cd: Change Directory



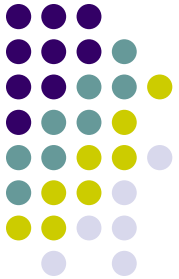
- "cd *dir*": Change the current directory to *dir*.
- The variable **HOME** is the default *dir*.
  - Note that $HOME has been setup automatically when you login the system

$PATH: A Very Important Shell Variable



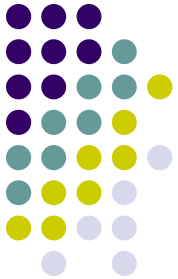
- $PATH sets the search paths of commands
  - Shell will search for a command only in those paths listed in the $PATH
    - Commands are searched in the listed path order
    - Only the first found command will be returned.
    - If commands are not in paths of $PATH, they will not be found
      - For example, commands in the current dir (marked by ".") will be run by "`./executable_program`"
  - This is the single shell variables troubles most new users

$PATH: Example and Setup Files



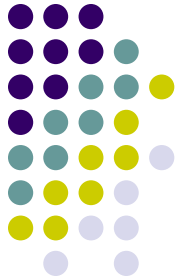
- bash
  - PATH="/usr/bin:/bin:/usr/local/bin:${PATH}"
- tcsh
  - setenv PATH="/usr/bin:/bin:/usr/local/bin:${PATH}"

Shell Keyboard Shortcuts



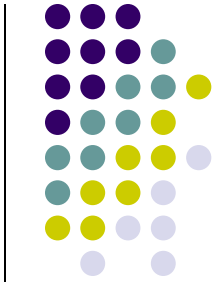
- Command history
  - View previous used commands with ↑↓keys.
  - The ← and → arrow keys move the cursor left and right on the current line, allowing you to edit your commands.
  - Ctrl+A (Ctrl+E) bring you to the begin (the end) of the current line.
  - The Backspace (Ctrl+H) and Del (Ctrl+D) keys work as expected.
  - Ctrl+K will delete from the position of the cursor to the end of line
  - Ctrl+W will delete the word before the cursor.
- Other convenient shell short-cuts
  - Ctrl+D on a blank line == exit.
  - Ctrl+C will interrupt the currently running command, or if you were editing your command line, it will cancel the editing and get you back to the prompt.
  - Ctrl+L clears the screen.
  - Ctrl+S and Ctrl+Q, which are used to suspend and restore output to the screen. They are not used very often, but you might suspend the current session by accidentally type Ctrl+S. Then try Ctrl+Q to return you the screen.

Shell Command Line Completion



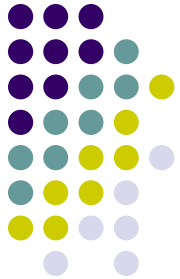
- Reduce the typed character for users
- Example:
  - Assume we have two files in the current directory: file_with_very_long_name_impossible_to_type and file_of_shorter_names
  - $ less fi<TAB>
  - $ less file_
  - less file_w<TAB>
  - less file_with_very_long_name_impossible_to_type

Revisiting ls



- -a lists all files, including **hidden files** Remember that in UNIX , hidden files are those whose names begin with a .;
- -A lists “almost” all files, which means every file the -a option would print except for “.” and “..”
- -R: lists recursively, i.e. all files and subdirectories of directories entered on the command line.
- -s: prints the file size in kilobytes next to each file.
- -l: prints additional information about the files.
- -d: treats directories on the command line as if they were normal files rather than listing their contents.

More File Handling Utilities

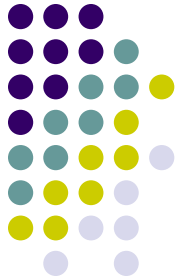


- mkdir: Creating Empty Directories
  - -p option.
    - Create parent directories if they did not exist before. Without this option, mkdir would just fail, complaining that the parent directories do not exist
    - Return silently if the directory which you want to create already exists. Similarly, without the -p option, mkdir will send back an error message, complaining that the directory already exists.
  - Example:
    - mkdir new_dir
    - mkdir –p new_parent/new_dir
- touch: Creating Empty Files
  - Example:
    - touch new_file

More File Handling Utilities (Cont.)

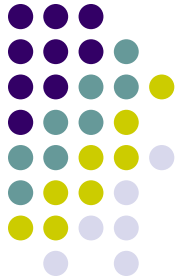
- rm: Deleting Files or Directories
  - -r, or -R: delete recursively. This option is **mandatory** for deleting a directory, empty or not. However, you can also use rmdir to delete empty directories.
  - -i: request confirmation before each deletion. rm is usually an **alias** to rm -i, for safety reasons.
  - -f, the opposite of -i, forces deletion of the files or directories, even if the user has no write access on the files.
  - Example
    - rm -i images/*.jpg
    - rm -Rf images/misc/ file*:
- mv: Moving or Renaming Files
  - -f: forces operation – no warning if an existing file is overwritten.
  - -i: the opposite. Asks the user for confirmation.
  - -v: **verbose** mode, report all changes and activity.
- cp: Copying Files and Directories
  - -R, -i, -f, -v

Viewing File Contents



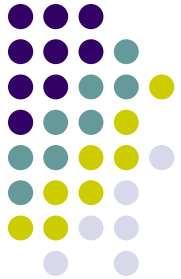
- General purpose editors like gvim or gedit are used, but the following are useful for quick checks (and pipe, too)
- cat
  - to print file contents on the standard outputs
- tail/head
  - print the last (first) 10 lines of a file to std outputs.
  - tail /var/log/mail/info
  - tail –n20 /var/log/mail/info (print the last 20 lines)
  - tail –f /var/log/messages (print to stdio whenever lines are added to the file)
- less is more
  - less and more to print file contents on stdio page by page (page size is defined by the current terminal)
  - a GNU program for traditional UNIX “more”;
  - It can go forward and backward.
  - less starts without reading the entire file; indispensible for viewing large files
  - Use “q” for quit and “h” for help.

Analyzing and Finding Files



- grep: Locate Strings in Files
- fgrep: Locate Strings in Files with Patterns Listed in another file
- wc: Calculating Elements in Files
- sort: Sorting File Content
- find: Find Files According to Certain Criteria

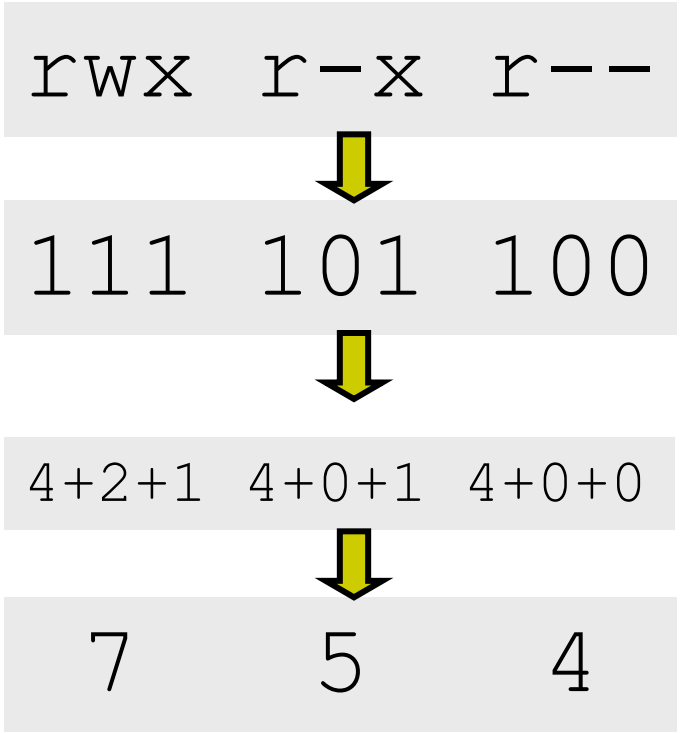
Changing File Attributes



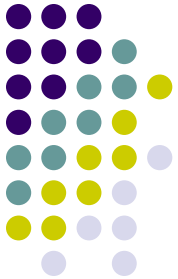
- chmod: Changing Permissions on Files and Directories
- Two forms of specifying options
  - in octal number
  - in character expression

chmod in octal

- The 9 permission characters in octal numbers
- Example:
- To change a file permission to rwxr-xr--, use “chmod 754 filename”

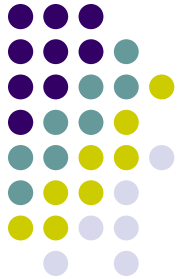


File Archiving and Data Compression with tar



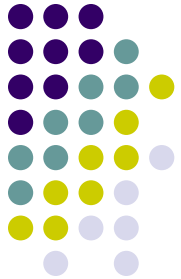
- $ tar cjf ~/images.tar.bz2 images/
  - *to create (c option) the "~/images.tar.bz2" file (follow by f) with compression (j option) from the "images/" directories.*
- $ tar tjvf images.tar.bz2
  - *to test (t option) and print (v option) the files/directories in the "images.tar.bz2" file (follow by f) with decompression (j option).*
- $ tar xjvf images.tar.bz2
  - *to extract (x option) and print (v option) the files/directories in the "images.tar.bz2" file (follow by f) with decompression (j option).*
  - *Note that this will overwrite files/directories in the current dir.*
- $ tar xzvf images.tar.gz
- $ tar xzvf images.tgz

Data Compression and Decompression with gzip/bzip2



- gzip and bzip2
- options
  - -1, ..., -9: set the compression ratio. The higher the number, the better the compression, but better also means slower.
  - -d: uncompress file(s). This is equivalent to using gunzip or bunzip2.
  - -c: dump the result of compression/decompression of files given as parameters to the standard output.
- By default, both gzip and bzip2 erase the file(s) that they have compressed (or uncompressed)
  - bzip2 has a –k option to avoid this
- Examples
  - $ bzip2 -9 afile.txt
    - At the output, there will be afile.txt.gz file (no more afile.txt).
  - bzip2 -dc images.tar.bz2 | gzip -9 >images.tar.gz
    - Recompression using gzip format

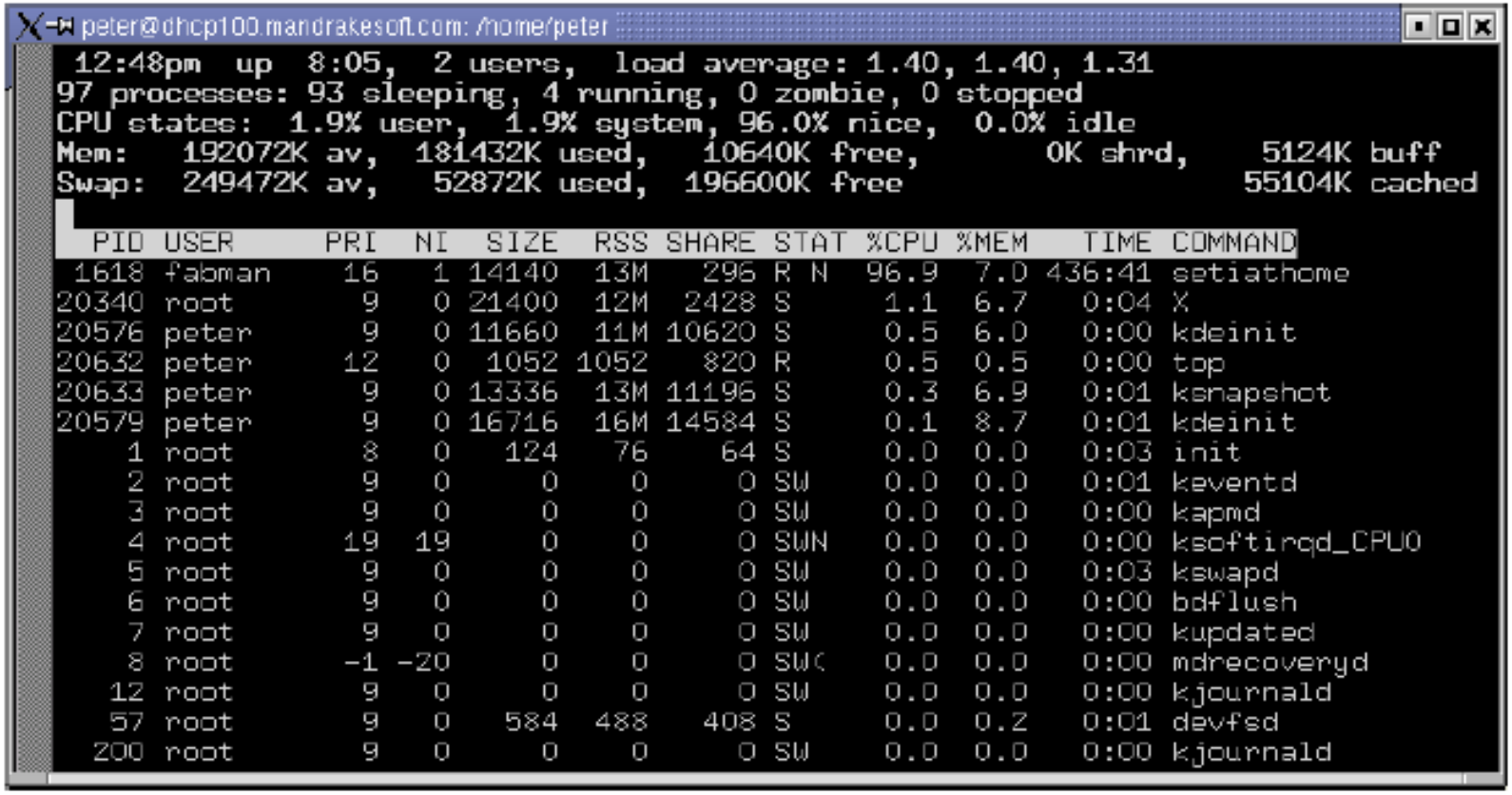
Listing Processes



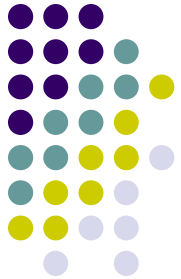
- ps
  - BSD and SVR4 has totally different options
- Fortunately you don't care too much about many other options
  - ps aux | less
  - ps –ef | less
- ps options
  - a: also displays processes started by other users;
  - x: also displays processes with no control terminal or with a control terminal different to the one you are using;
  - u: displays for each process the name of the user who started it and the time at which it was started.

Viewing Processes

- top



Sending Signals to Processes



- kill
- Example
  - $ kill -9 785
  - Send a KILL signal to process 785