



## EE6094 CAD for VLSI Design



# Workstation User's Guide

Andy, Yu-Guang Chen  
Assistant Professor, Department of EE  
National Central University  
andyygchen@ee.ncu.edu.tw

Slides Credit: TA Shih-Cheng Huang / TA Yi-Ting Lin



2024/2/29

Andy Yu-Guang Chen

1



## Outline




- ◆ Workstation
- ◆ Vim
- ◆ Tmux




2024/2/29

Andy Yu-Guang Chen


2




# Outline




- ◆ Workstation
  - ◆ Vim
  - ◆ Tmux



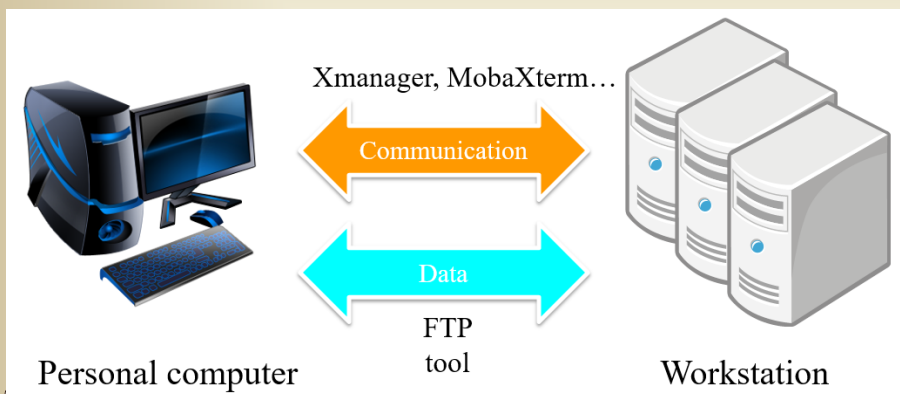
2024/2/29 Andy Yu-Guang Chen 3



# Workstation



- ◆ Introduction of workstation



Personal computer


Workstation

Xmanager, MobaXterm...

Communication

Data

FTP tool



2024/2/29 Andy Yu-Guang Chen 4



# Workstation



- ◆ You are requested to complete most of your programming assignments on our server
  - And we will evaluate your program with the server
- ◆ Server setting
  - System: CentOS 7.1.1503
  - gcc version: 6.2.0
  - Host: 140.115.71.192
  - Port: 22 ( Please use SSH to connect to the machine )
    - Default account: s(Your Student\_ID)
      - Ex. s123456789
    - Default password: Your Student\_ID
      - Ex. 123456789



2024/2/29

Andy Yu-Guang Chen

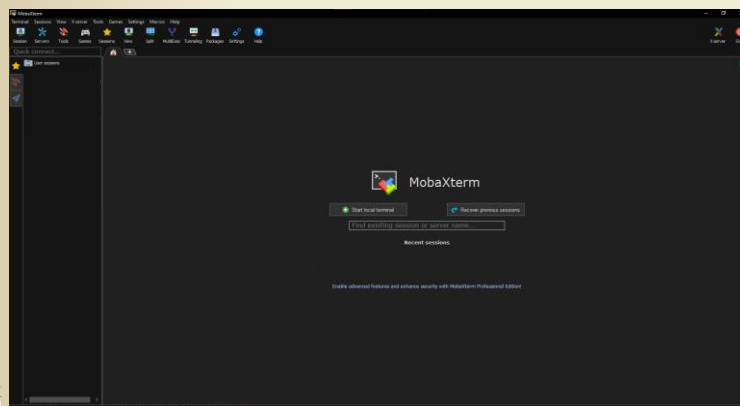
5



# Workstation



- ◆ How to connect to a server with SSH Client?
  - MobaXterm



2024/2/29

Andy Yu-Guang Chen

6



# Workstation



## ◆ MobaXterm

- MobaXterm is your ultimate toolbox for remote computing.
- MobaXterm provides all the important remote network tools (SSH, X11, RDP, VNC, FTP, MOSH, ...) and Unix commands (bash, ls, cat, sed, grep, awk, rsync, ...) to Windows desktop
- There are many advantages of having an All-In-One network application for your remote tasks



2024/2/29

Andy Yu-Guang Chen

7

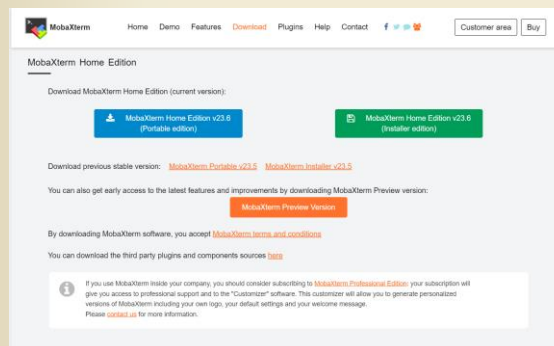


# Workstation



## ◆ How to connect the Linux server

- Step 1: download MobaXterm  
<https://mobaxterm.mobatek.net/download-home-edition.html>



2024/2/29

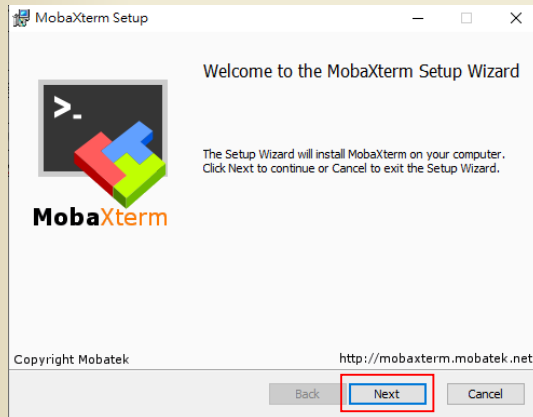
Andy Yu-Guang Chen

8

# Workstation

## ◆ How to connect Linux server

### ➤ Step 2-1: install MobaXterm



2024/2/29

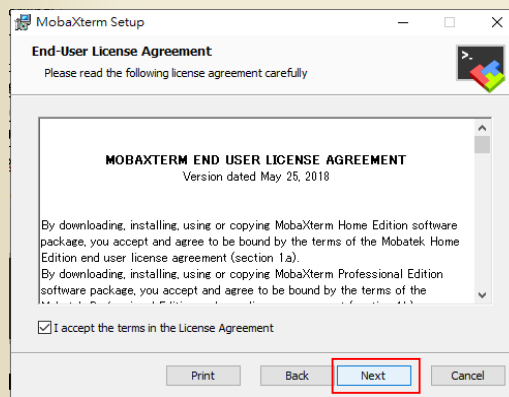
Install MobaXterm with following steps

9

# Workstation

## ◆ How to connect Linux server

### ➤ Step 2-2: install MobaXterm



2024/2/29

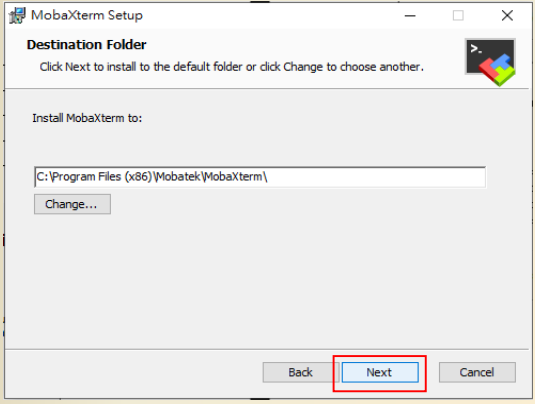
Install MobaXterm with following steps

10

# Workstation

◆ How to connect Linux server

➤ Step 2-3: install MobaXterm



MobaXterm Setup

**Destination Folder**  
Click Next to install to the default folder or click Change to choose another.

Install MobaXterm to:

C:\Program Files (x86)\Mobatek\MobaXterm\

Change...

Back Next Cancel

2024/2/29

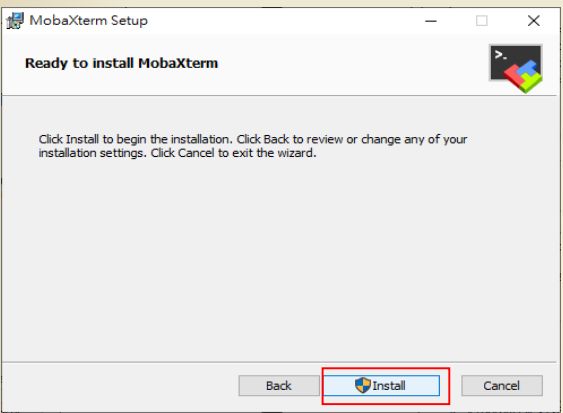
Install MobaXterm with following steps

11

# Workstation

◆ How to connect Linux server

➤ Step 2-4: install MobaXterm



MobaXterm Setup

**Ready to install MobaXterm**

Click Install to begin the installation. Click Back to review or change any of your installation settings. Click Cancel to exit the wizard.

Back Install Cancel

2024/2/29

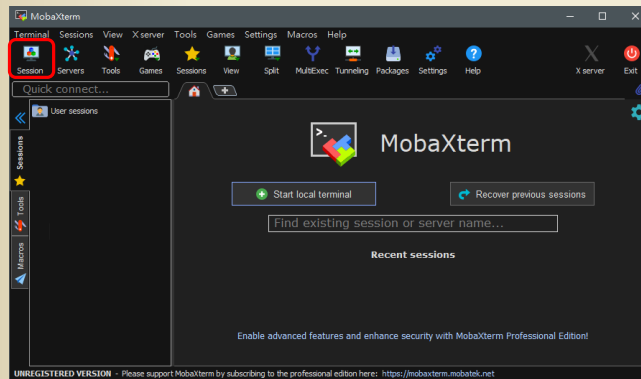
Install MobaXterm with following steps

12

# Workstation

## ◆ How to connect Linux server

➤ Step 3-1: Click session to create a new session



2024/2/29

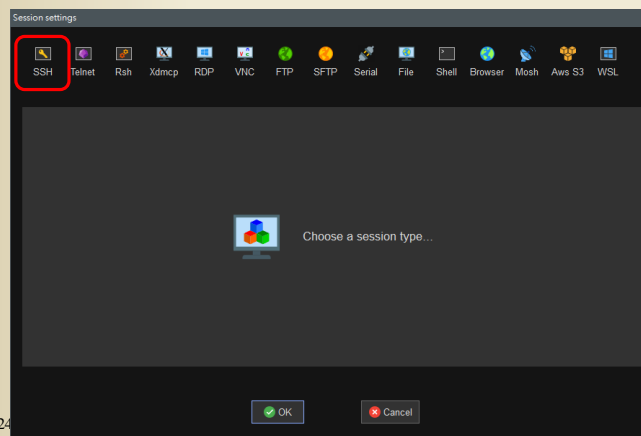
Andy Yu-Guang Chen

13

# Workstation

## ◆ How to connect Linux server

➤ Step 3-2: Click SSH to create a new SSH session



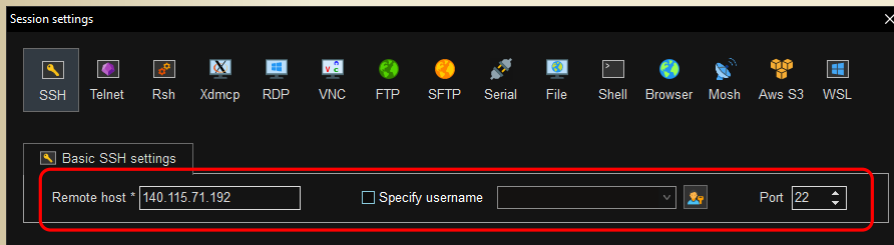
2024

14

# Workstation

## ◆ How to connect Linux server

- Step 3-3: Fill in the given Host to the Host field and set the port number to 22



2024/2/29

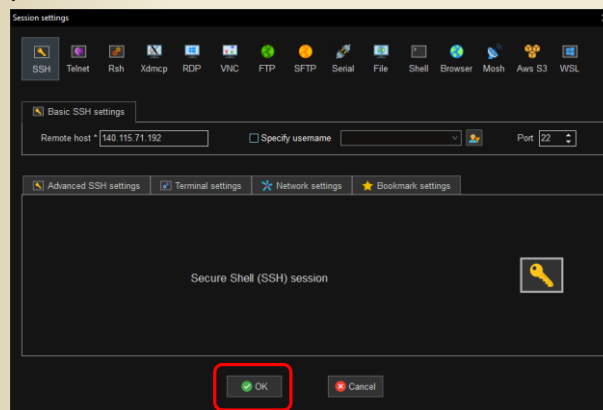
Andy Yu-Guang Chen

15

# Workstation

## ◆ How to connect Linux server

- Step 3-4: Click ok



2024/2/29

Andy Yu-Guang Chen

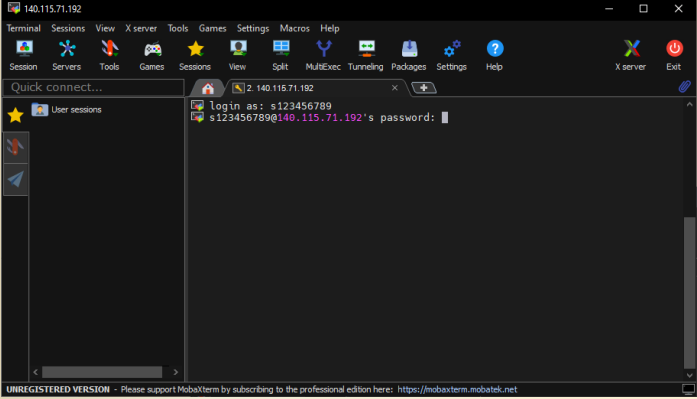
16



# Workstation

◆ How to connect Linux server

➤ Step 3-5: Login

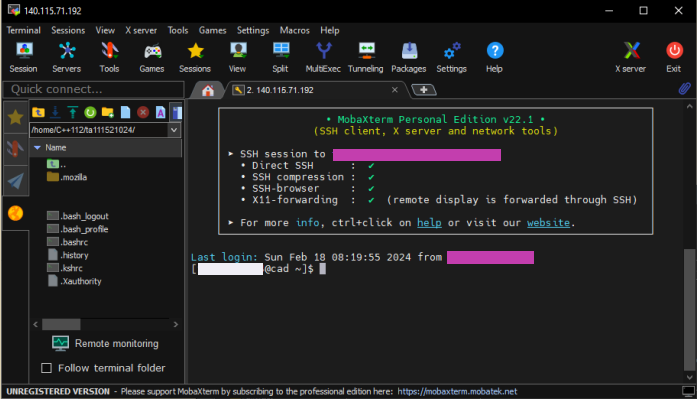


2024/2/29 Andy Yu-Guang Chen 17

# Workstation

◆ How to connect Linux server

➤ Step 3-6: Login



2024/2/29 Andy Yu-Guang Chen 18



# Workstation



## ◆ Change your workstation password

- Step 4-1: Key in "passwd"
- Step 4-2: Key in your current password
- Step 4-3: Key in your new password
- Step 4-4: Retype your new password again
- The info with "successfully" means password change is success

```
[_____]@cad ~]$ passwd Step4.1
Changing password for user _____.
Changing password for _____.
(current) UNIX password: Step4.2
New password: Step4.3
Retype new password: Step4.4
passwd: all authentication tokens updated successfully.
```



2024/2/29

Andy Yu-Guang Chen

19



# Basic command of workstation



- |                                 |                            |
|---------------------------------|----------------------------|
| ◆ ls (list)                     | ◆ mkdir (make directory)   |
| ◆ ll (long list format)         | ◆ rmdir (remove directory) |
| ◆ cd (change directory)         | ◆ tar (compression tool)   |
| ◆ pwd (print working directory) | ◆ passwd (password)        |
| ◆ cp (copy)                     | ◆ Ctrl + c (force quit)    |
| ◆ mv (move)                     | ◆ ps (process status)      |
| ◆ rm (remove)                   | ◆ kill (kill process)      |



2024/2/29

Andy Yu-Guang Chen

20



# Basic Command



## ◆ mkdir

```
[ ]@cad ~]$ mkdir PA1
[ ]@cad ~]$
```

## ◆ ls

```
[ ]@cad ~]$ ls
[ ] PA1 PA2 PA3 PA4
```

## ◆ cd

```
[ ]@cad ~]$ cd PA1
[ ]@cad ~/PA1]$
```



2024/2/29

Andy Yu-Guang Chen

21



# Basic Command



## ◆ cp

```
[ ]@cad ~/PA1]$ cp ../test.cpp PA1.cpp
```

## ◆ rm

```
[ ]@cad ~/PA1]$ ls
[ ] PA1.cpp
[ ]@cad ~/PA1]$ rm PA1.cpp
[ ]@cad ~/PA1]$ ls
[ ]@cad ~/PA1]$
```



2024/2/29

Andy Yu-Guang Chen

22



# Basic Command



## ◆rmkdir

```
[ ]@cad ~]$ ls
[ ] PA1 PA2 PA3 PA4 test.cpp
[ ]@cad ~]$ rmdir PA1
[ ]@cad ~]$ ls
[ ] PA2 PA3 PA4 test.cpp
[ ]@cad ~]$
```



2024/2/29

Andy Yu-Guang Chen

23



# Basic Command



## ◆mv


```
[ ]@cad ~/PA1]$ ls
README.md [ ]
[ ]@cad ~/PA1]$ ls ../
[ ] PA1 PA2 PA3 PA4 test.cpp
[ ]@cad ~/PA1]$ mv ../test.cpp ./
[ ]@cad ~/PA1]$ ls
README.md test.cpp
[ ]@cad ~/PA1]$ ls ../
[ ] PA1 PA2 PA3 PA4 [ ]
[ ]@cad ~/PA1]$
```




2024/2/29

Andy Yu-Guang Chen


24




# Outline




- ◆ Workstation
- ◆ Vim
- ◆ Tmux



2024/2/29 Andy Yu-Guang Chen 25




# Vim




- ◆ It is an efficient text editor especially developed for Linux users, that it is mainly used to edit or create different types of files
- ◆ Vim is the most popular and extremely powerful text editor
  - It possesses a lot of features that you would not expect to have in a text editor

```
[ ]@cad ~/PA1]$ vim parser.cpp
```



Open parser.cpp or create a new file named parser.cpp



2024/2/29 Andy Yu-Guang Chen 26



# Vim



- ◆ The Vim editor is a modal text editor
  - it uses modes for different purposes like inserting text, running commands, and selecting text
- ◆ There are three modes of operation
  - Normal
    - The initial mode of the Vim editor
    - Normal mode is also known as command mode because all the keystrokes you perform are interpreted as commands
  - Insert
    - Insert mode is where you can insert your text in the file
    - This mode inserts every character you type at the current cursor location
  - Visual
    - Visual mode allows you to select text so that you may perform certain operations (cut, copy, delete) on it



2024/2/29

Andy Yu-Guang Chen

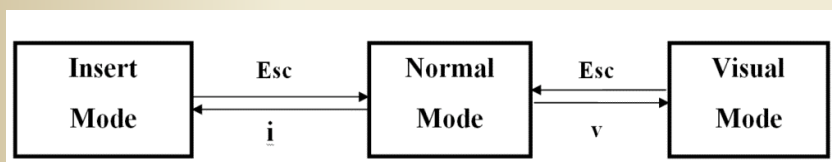
27



# Vim




## ◆ Changing the modes




2024/2/29

Andy Yu-Guang Chen

28




# Vim




◆ Normal mode

- You will see the below screen after executing the command
- This is your normal mode in Vim


```
~
~
~
"parser.cpp" [New File]
```



2024/2/29 Andy Yu-Guang Chen 29




# Vim



◆ Insert mode

- You should be in the Insert mode if you want to **edit** your file
- Press “i”, “a” or “o” from your keyboard, and you will be in insert mode
- Press Esc to back to normal mode

```
#include <string>
void main() {
}
~
-- INSERT --
```



2024/2/29 Andy Yu-Guang Chen 30



# Vim



## ◆ Saving your work

- When you are in normal mode, press “:w” to save your work and press “:q” to exit vim
- You also can use “:wq” to save and exit vim

```
#include <string>

void main() {
~
:wq
```



2024/2/29

Andy Yu-Guang Chen

31



# Vim



## ◆ File related commands

:w	write the file to the disk
:q	quit vi without saving the file
:wq	write the file to disk and quit vi
:q!	ignore the warning and discard the change
:w filename	save the file as filename



2024/2/29

Andy Yu-Guang Chen

32





# Vim



## ◆ Moving the cursor

j	move the cursor down one line
k	move the cursor position up one line
l	move the cursor to the bottom of the screen
0	move to the beginning of the line
\$	move to the end of the line



2024/2/29

Andy Yu-Guang Chen

33



# Vim



## ◆ Inserting Text

I	insert text at the beginning of the line
i	insert text before the current cursor location
a	insert text after the current cursor location
o	create a new line for the text below the current cursor location
O	create a new line for text above the current cursor location



2024/2/29

Andy Yu-Guang Chen

34



# Vim



## ◆ Changing text

cc	remove the whole line and start Insert mode
s	remove the character under the cursor and start Insert mode
r	replace the character under the cursor



2024/2/29

Andy Yu-Guang Chen

35



# Vim



## ◆ Copying pasting

y	copy the selected text to clipboard
yy	copy current line
P	insert the text "before" the cursor
p	insert the text at the point after the cursor



2024/2/29

Andy Yu-Guang Chen

36



# Vim



## ◆Deleting Text

X	delete the character before the current location
x	delete the character under the current location
D	cut to the end of line
dd	cut current line



2024/2/29

Andy Yu-Guang Chen

37



# Vim



## ◆Undo/Redo

u	undo last change
Ctrl+R	redo



2024/2/29

Andy Yu-Guang Chen

38



# Vim



version 1.1  
April 1st, 06

## vi / vim graphical cheat sheet

Esc  
normal mode

~ toggle case	! external filter	@ play macro	# prev ident	\$ col	% goto match	^ "soft" bol	& repeat	* next ident	( begin sentence	) end sentence	"soft" bol down	+ next line
1 goto mark	2	3	4	5	6	7	8	9	0 "hard" bol	- prev line	= auto-format	
Q ex mode	W next word	E end word	R replace mode	T back-till	Y yank line	U undo line	I insert at bol	O open above	P paste before	{ begin parag.	} end parag.	
q record macro	w next word	e end word	r replace char	t till	y yank	u undo	i insert mode	o open below	p paste after	* misc	* misc	
A append at eol	S subst line	D delete to eol	F "back" find ch	G eof	H screen top	J join lines	K help	L screen bottom	: ex cmd line	" reg. spec	hol/ goto col	
a append	s subst char	d delete	f find char	g extra cmds	h ←	j ↓	k ↑	l →	: repeat v/t/T/F	' goto mk. bol	* not used!	
Z quit	X back-space	C change to eol	V visual lines	B prev word	N prev (find)	M screen mid	< un-indent	> indent	? find (rev.)			
Z extra cmds	X delete char	C change	V visual mode	b prev word	n next (find)	m set mark	> reverse	> repeat cmd	/ find			

**motion** moves the cursor, or defines the range for an operator

**command** direct action command, if **red**, it enters insert mode

**operator** requires a motion afterwards, operates between cursor & destination

**extra** special functions, requires extra input

**Q** commands with a dot need a char argument afterwards

bol = beginning of line, eol = end of line, mk = mark, yank = copy

words: `quux(foo, bar, baz)`

WORDS: `quux(foo, bar, baz)`

**Main command line commands ('ex'):**

:w (save), :q (quit), :q! (quit w/o saving)

:e f (open file f),

:%s/x/y/g (replace 'x' by 'y' filewide),

:h (help in vim), :new (new file in vim),

**Other important commands:**

CTRL-R: redo (vim),

CTRL-F/-B: page up/down,

CTRL-E/-Y: scroll line up/down,

CTRL-V: block-visual mode (vim only)

**Visual mode:**

Move around and type operator to act on selected region (vim only)

**Notes:**

- (1) use "x before a yank/paste/del command to use that register ('clipboard') (x=a..z,') (e.g.: "ay\$ to copy rest of line to reg 'a')
- (2) type in a number before any action to repeat it that number of times (e.g.: 2p, d2w, 5l, d4j)
- (3) duplicate operator to act on current line (dd = delete line, >> = indent line)
- (4) ZZ to save & quit, ZQ to quit w/o saving
- (5) zt: scroll cursor to top, zb: bottom, zz: center
- (6) gg: top of file (vim only), gf: open file under cursor (vim only)

For a graphical vi/vim tutorial & more tips, go to [www.viemu.com](http://www.viemu.com) - home of ViEmu, vi/vim emulation for Microsoft Visual Studio



# Vim



version 1.1  
April 1st, 06  
翻譯: 2006-5-24

## vi / vim 圖解鍵盤指令

Esc  
命令模式

~ 轉換大小寫	! 外部過濾器	@ 播放宏	# 前一個標識符	\$ 行末	% 跳至匹配	^ (軟) 行首	& 重複	* 下一個標識符	( 句首	) 句尾	"soft" bol 下一句首	+ 下行
1 跳至標記	2	3	4	5	6	7	8	9	0 (硬) 行首	- 上一句首	= 自動縮排	
Q 切換到 ex 模式	W 下一單詞	E 詞尾	R 替換模式	T back-till	Y 複製行	U 回復行	I 到行首插入	O 到行尾插入	P 貼上(前)	p 貼上(後)	{ 段首	}
q 複製單字	w 下一單詞	e 詞尾	r 替換字元	t till	y 複製	u 回復	i 插入模式	o 到段尾	P 貼上(前)	p 貼上(後)	* 雜項	
A 在行末附加	S 刪除字元並輸入	D 刪除到行尾	F 到前字元尋找	G 到行尾附加	H 螢幕頂端	J 合併行	K 幫助畫面	L 螢幕底端	: 命令	" 暫存寄存器	hol/ 跳至列	
a 附加	s 刪除字元並輸入	d 刪除	f 到字元尋找	g 附加命令	h ←	j ↓	k ↑	l →	: repeat v/t/T/F	' 跳至標記	* 未使用	
Z 退出	X 刪除	C 修改至行末	V 視覺行模式	B 前一個單詞	N 前一個(尋找)	M 螢幕中間	< 反縮排	> 縮排	? 尋找(反)			
Z 附加命令	X 刪除字元	C 修改	V 視覺模式	b 前一個單詞	n 下一個(尋找)	m 設定標記	> 反向	> 重複命令	/ 尋找			

**動作** 移動游標, 或定義欲操作的範圍

**指令** 直接執行的指令

**操作** 後接用以表示操作範圍的指令

**extra** 特殊功能, 需額外輸入

**Q** 後接字元構成的參數

**主要 ex 指令:**

:w (儲存), :q (退出), :q! (不儲存退出)

:e f (開啟文件 f),

:%s/x/y/g (以 'y' 全文替換 'x'),

:h (輔助文件 in vim), :new (新建文件 in vim)

**其它重要指令:**

Ctrl-R: 重復 (vim),

Ctrl-F/-B: 向前(下)翻頁/向後(上)翻頁,

Ctrl-E/-Y: 向前(下)一列/向後(上)一列,

Ctrl-V: 切換 visual 模式 (vim only)

**visual 模式:**


游標移動選擇區域, 並執行特定操作 (vim only)

**備註:**


- (1) 在複製/貼上/刪除 指令前使用 "x (x=a..z,') 使用指令的暫存器 (如 "ay\$ 複製該行目前位置至行尾的內容到暫存器 a)
- (2) 命令前添加數字 重複指定次數的操作 (如: 2p, d2w, 5l, d4j)
- (3) 重複游標所在字元處指定的操作 (dd = 刪除本行, >> = 行首縮排)
- (4) ZZ 儲存離開, ZQ 不儲存離開
- (5) zt 移動游標所在行至畫面頂端, zb: 底端, zz: 中央
- (6) gg 文件開端 (vim only), gf: 開啟游標處的文件名稱 (vim only)

原圖: [www.viemu.com](http://www.viemu.com)


翻譯: fdi (Linuxsir), jserv




# Outline




- ◆ Workstation
- ◆ Vim
- ◆ Tmux



2024/2/29 Andy Yu-Guang Chen 41




# Tmux




- ◆ What is tmux?
  - Tmux is a terminal multiplexer that you can start a Tmux session and then open multiple windows inside that session
  - Session → windows


```
[ ]@cad ~/PA1]$ tmux
```



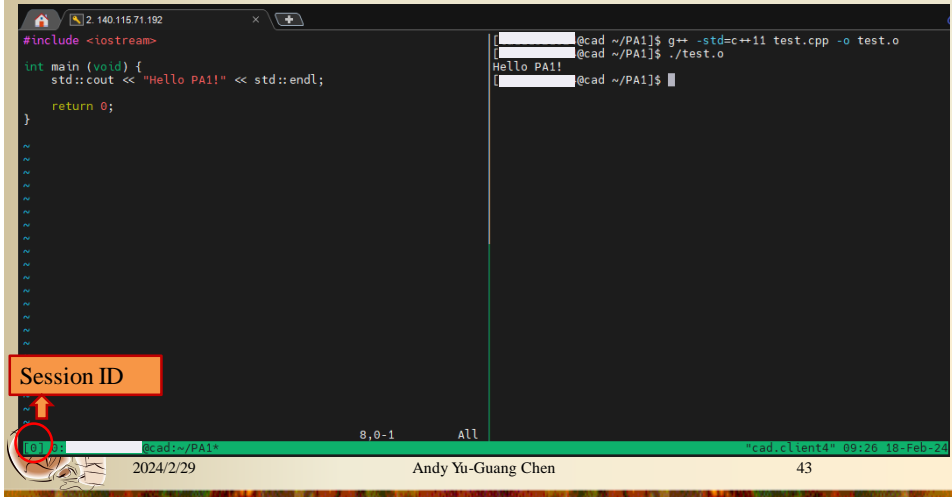
2024/2/29 Andy Yu-Guang Chen 42



# Tmux




◆ A tmux session with two windows




Session ID

2024/2/29 Andy Yu-Guang Chen 43




# Tmux



◆ Working with tmux windows and panes

- Ctrl+b c: Create a new window.
- Ctrl+b w: Choose window from a list.
- Ctrl+b 0: Switch to window 0.
- Ctrl+b n: Switch to next window.
- Ctrl+b p: Switch to previous window.
- Ctrl+b %: Split current pane horizontally into two panes.
- Ctrl+b |: Split current pane vertically into two panes.
- Ctrl+b arrow keys: Switch pane.



2024/2/29 Andy Yu-Guang Chen 44



# Reference

## ◆ Linux

- <https://files.fosswire.com/2007/08/fwunixref.pdf>
- <https://linux.vbird.org/>

## ◆ Vim

- <https://danielmiessler.com/study/vim/>
- <http://www.vixual.net/blog/archives/234>

## ◆ Tmux

- <https://blog.gtwang.org/linux/linux-tmux-terminal-multiplexer-tutorial/>
- <https://linuxize.com/post/getting-started-with-tmux/#starting-your-first-tmux-session>



2024/2/29

Andy Yu-Guang Chen

45